

Table II
Initial contents of Database D1

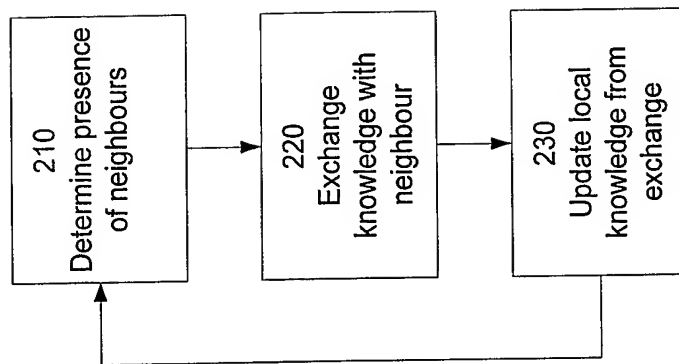
Rank	Node name	Best Neighbour
Ø	N1	N/A

Table III
Initial contents of Database D2

Rank	Node name	Best Neighbour
Ø	N2	N/A

Table IV
Initial contents of Database D3

Rank	Node name	Best Neighbour
Ø	N3	N/A



200

Fig. 2

Initial
Pass
through
Step
210

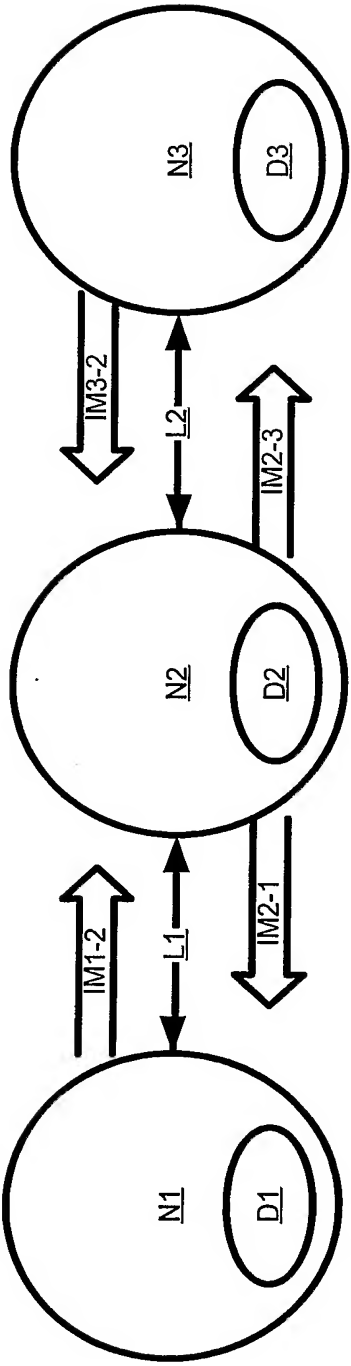


Table II
Initial contents of Database D1

Rank	Node name	Best Neighbour
Ø	N1	N/A

Table III
Initial contents of Database D2

Rank	Node name	Best Neighbour
Ø	N2	N/A

Table IV
Initial contents of Database D3

Rank	Node name	Best Neighbour
Ø	N3	N/A

Fig. 3

Initial
Pass
through
Step
220

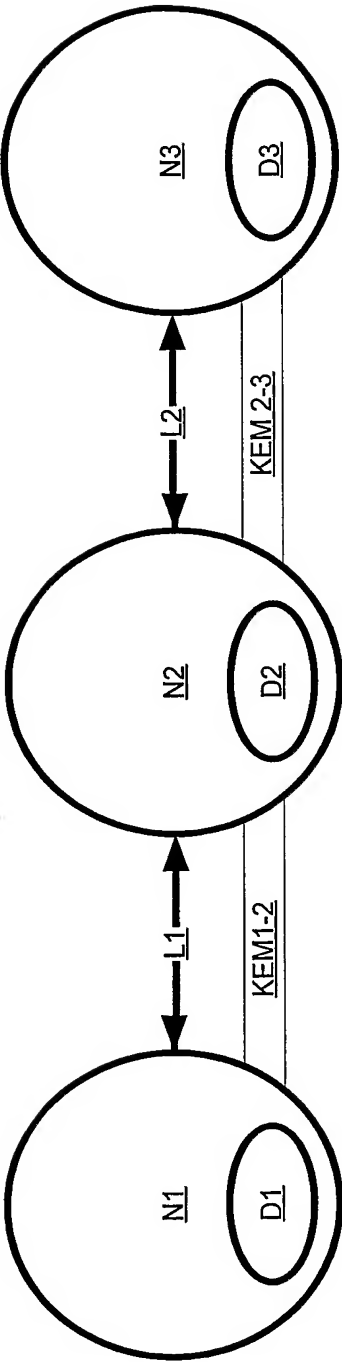


Table II
Initial contents of Database D1

Rank	Node name	Best Neighbour
Ø	N1	N/A

Table III
Initial contents of Database D2

Rank	Node name	Best Neighbour
Ø	N2	N/A

Table IV
Initial contents of Database D3

Rank	Node name	Best Neighbour
Ø	N3	N/A

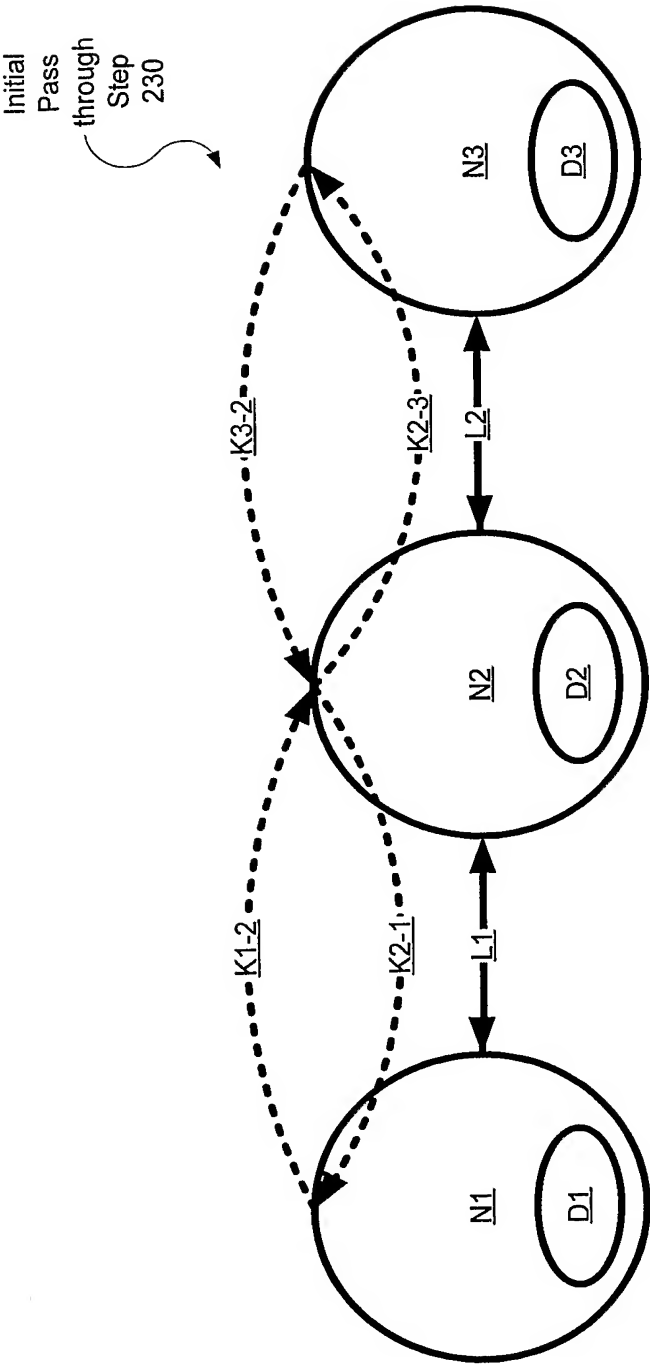


Table VII
Contents of Database D3

Rank	Node name	Best Neighbour
Ø	N3	N/A
1	N2	N2

Table VI
Database D2

Rank	Node name	Best Neighbour
Ø	N2	N/A
1	N1	N1
2	N3	N3

Table V
Database D1

Rank	Node name	Best Neighbour
Ø	N1	N/A
1	N2	N2

Fig. 5

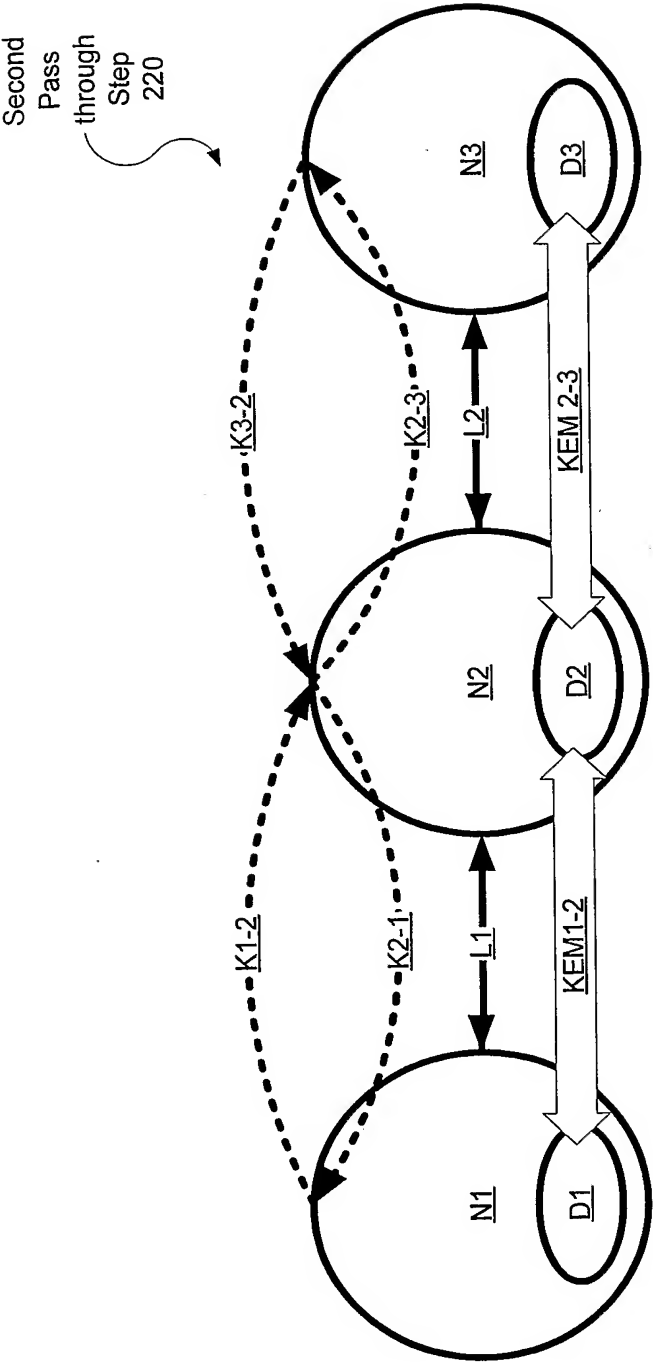


Table X
Contents of Database D3

Rank	Node name	Best Neighbour
0	N3	N/A
1	N2	N2
2	N1	N2

Table IX
Database D2

Rank	Node name	Best Neighbour
0	N2	N/A
1	N1	N1
2	N3	N3

Table VIII
Database D1

Rank	Node name	Best Neighbour
0	N1	N/A
1	N2	N2
2	N3	N2

Fig. 6

7/54

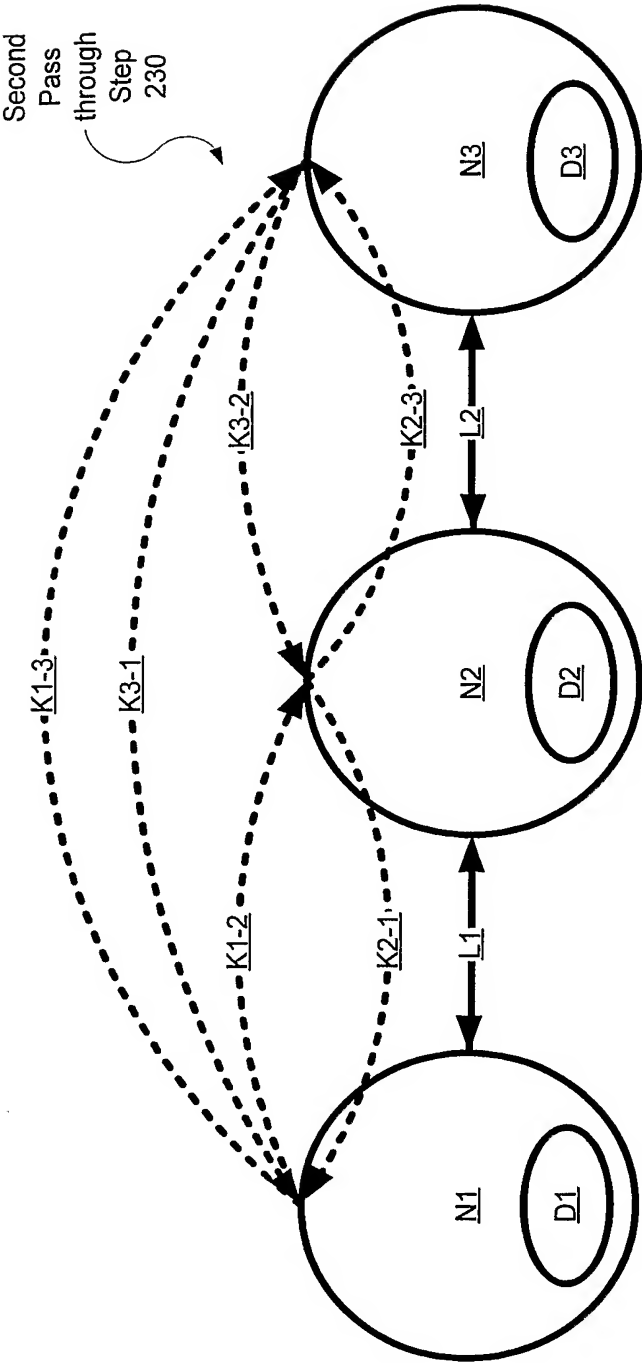


Table X
Contents of Database D3

Rank	Node name	Best Neighbour
Ø	N3	N/A
1	N2	N2
2	N1	N2

Table IX
Database D2

Rank	Node name	Best Neighbour
Ø	N2	N/A
1	N1	N1
2	N3	N3

Table VIII
Database D1

Rank	Node name	Best Neighbour
Ø	N1	N/A
1	N2	N2
2	N3	N2



Fig. 7

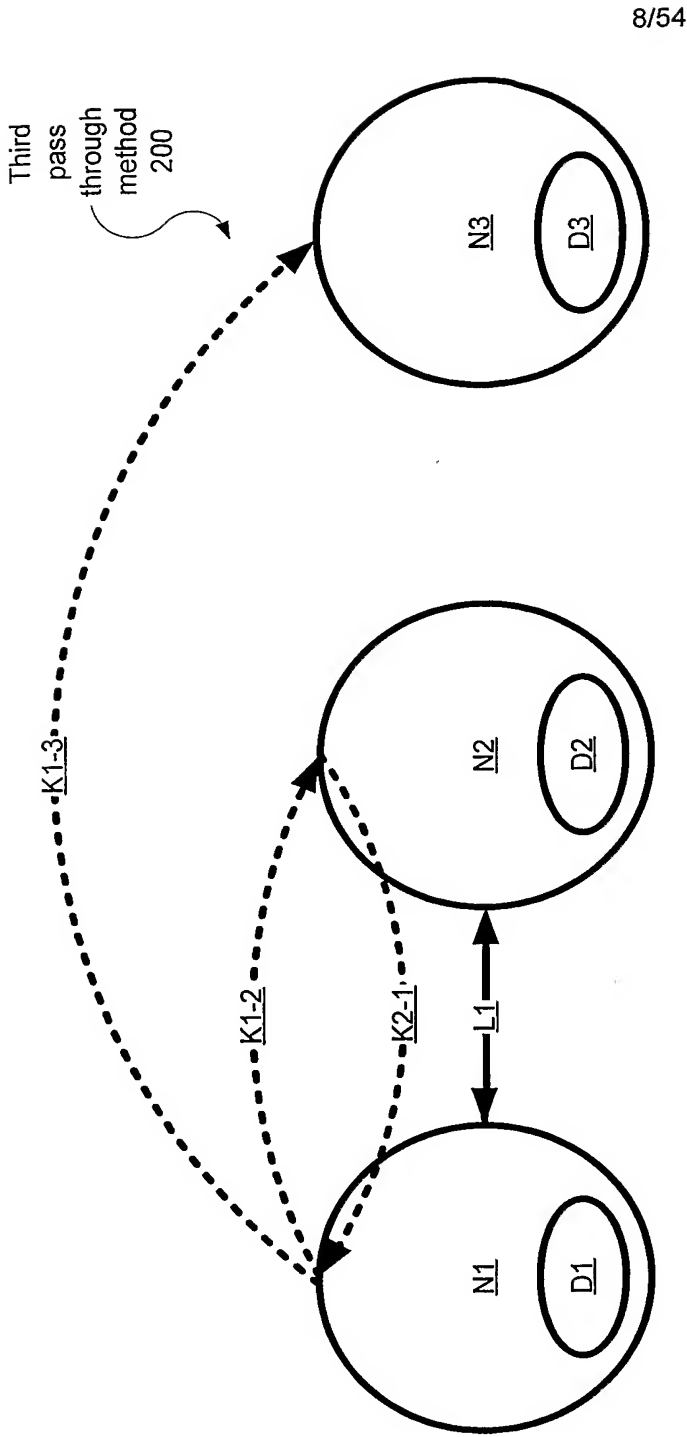


Table IV
Database D3

Rank	Node name	Best Neighbour
Ø	N3	N/A

Table XI
Database D2

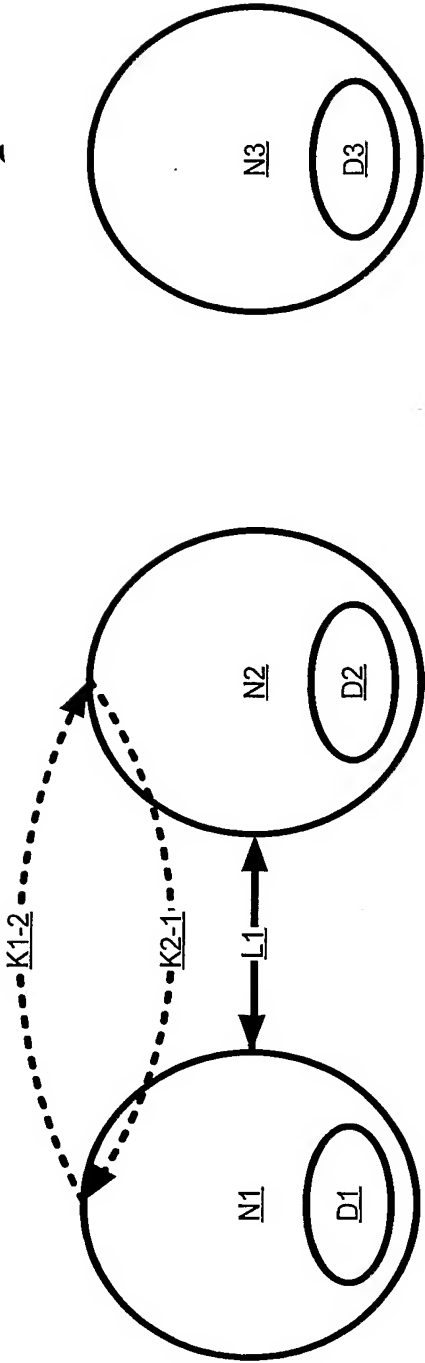
Rank	Node name	Best Neighbour
Ø	N2	N/A
1	N1	N1

Table VIII
Database D1

Rank	Node name	Best Neighbour
Ø	N1	N/A
1	N2	N2
2	N3	N2

Fig. 8

Fourth
pass
through
method
200



9/54

Table IV
Database D3

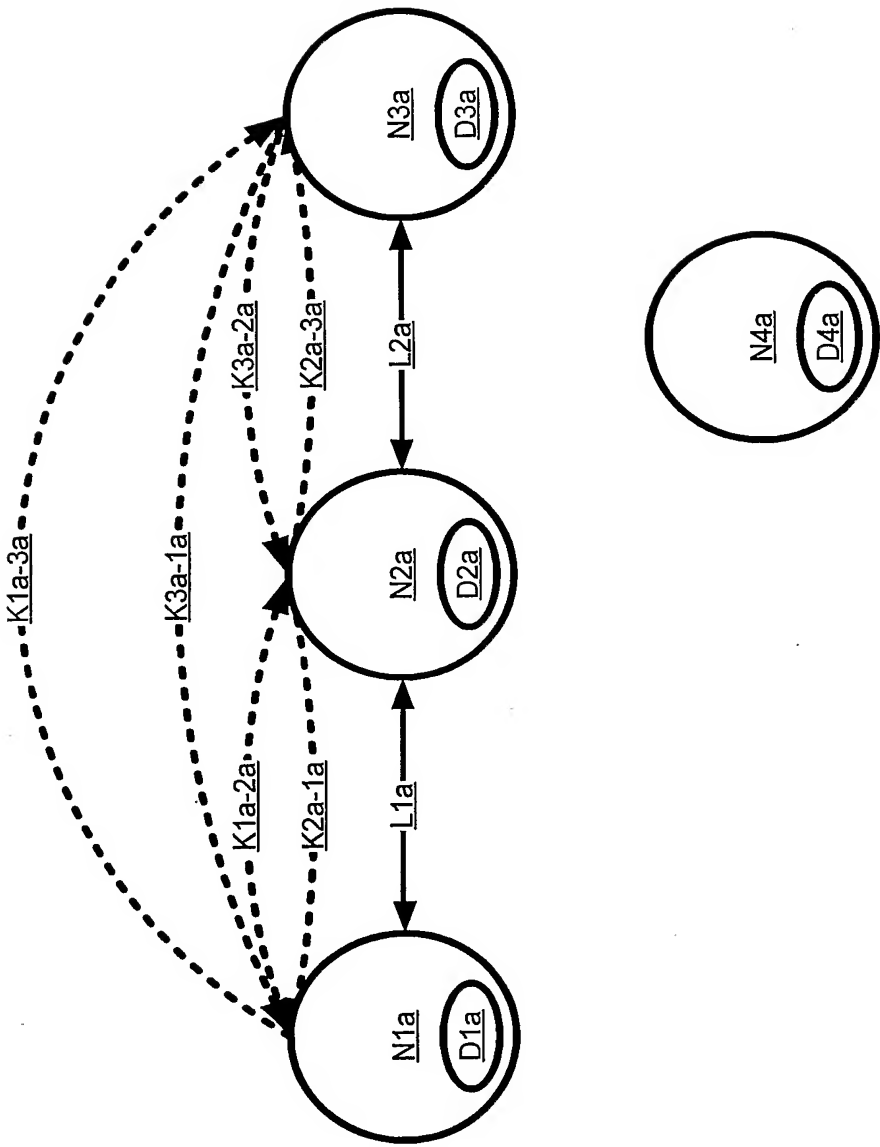
Rank	Node name	Best Neighbour
Ø	N3	N/A

Table XI
Database D2

Rank	Node name	Best Neighbour
Ø	N2	N/A
1	N1	N1

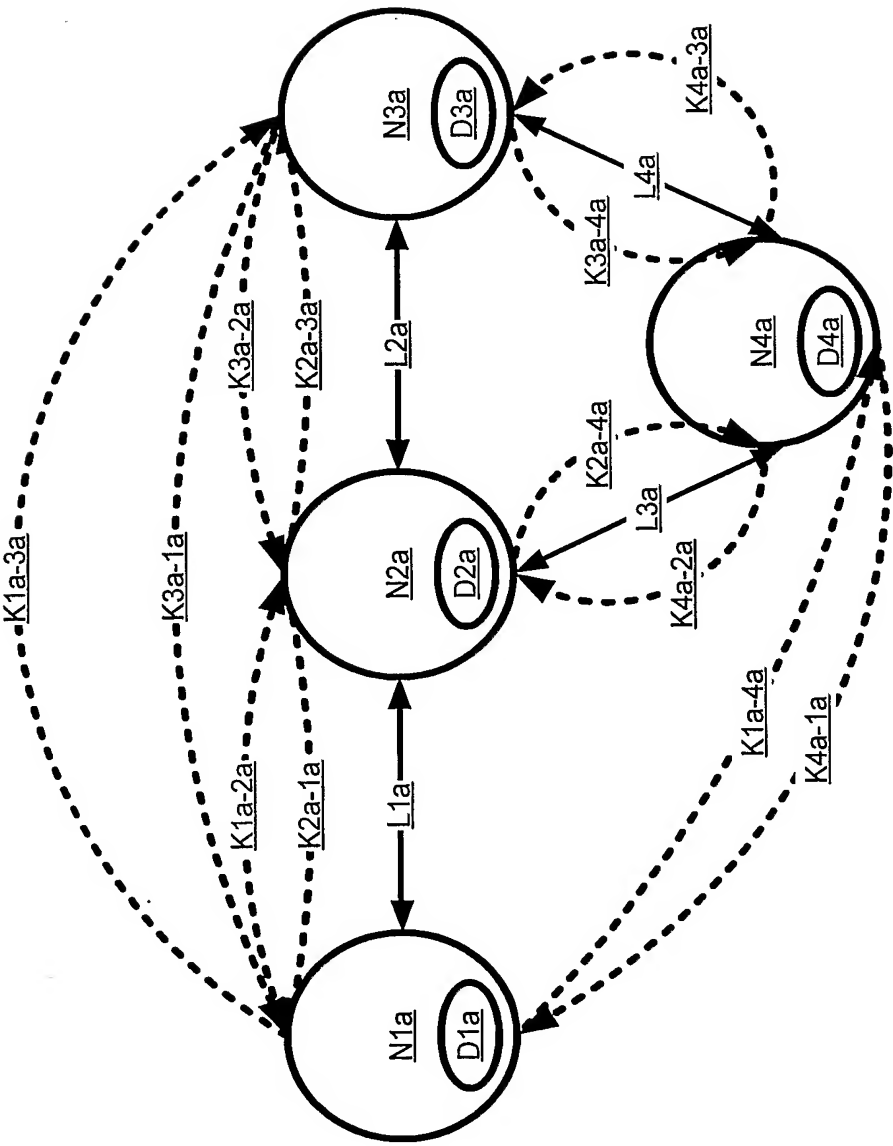
Table V
Database D1

Rank	Node name	Best Neighbour
Ø	N1	N/A
1	N2	N2



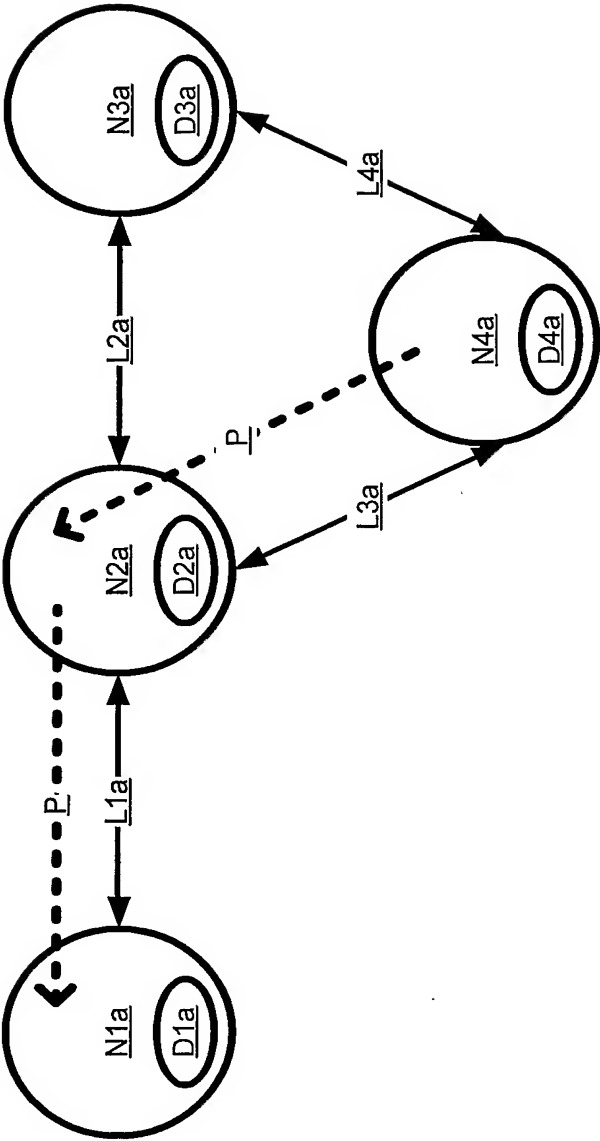
30a

Fig. 10



30a

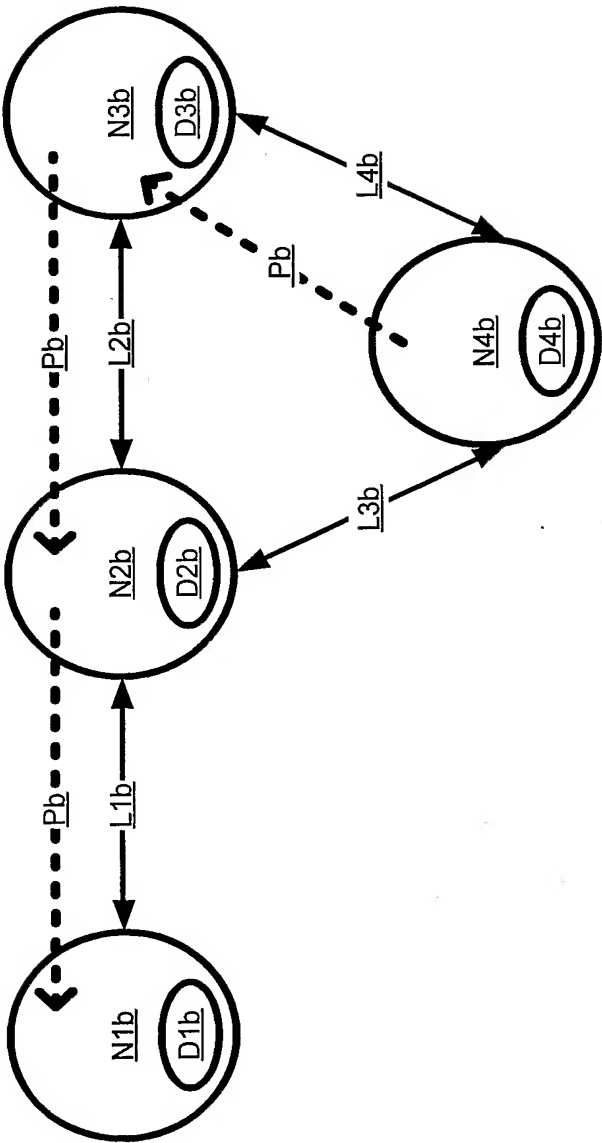
Fig. 11



30a

Fig. 12

Fig. 13



30b

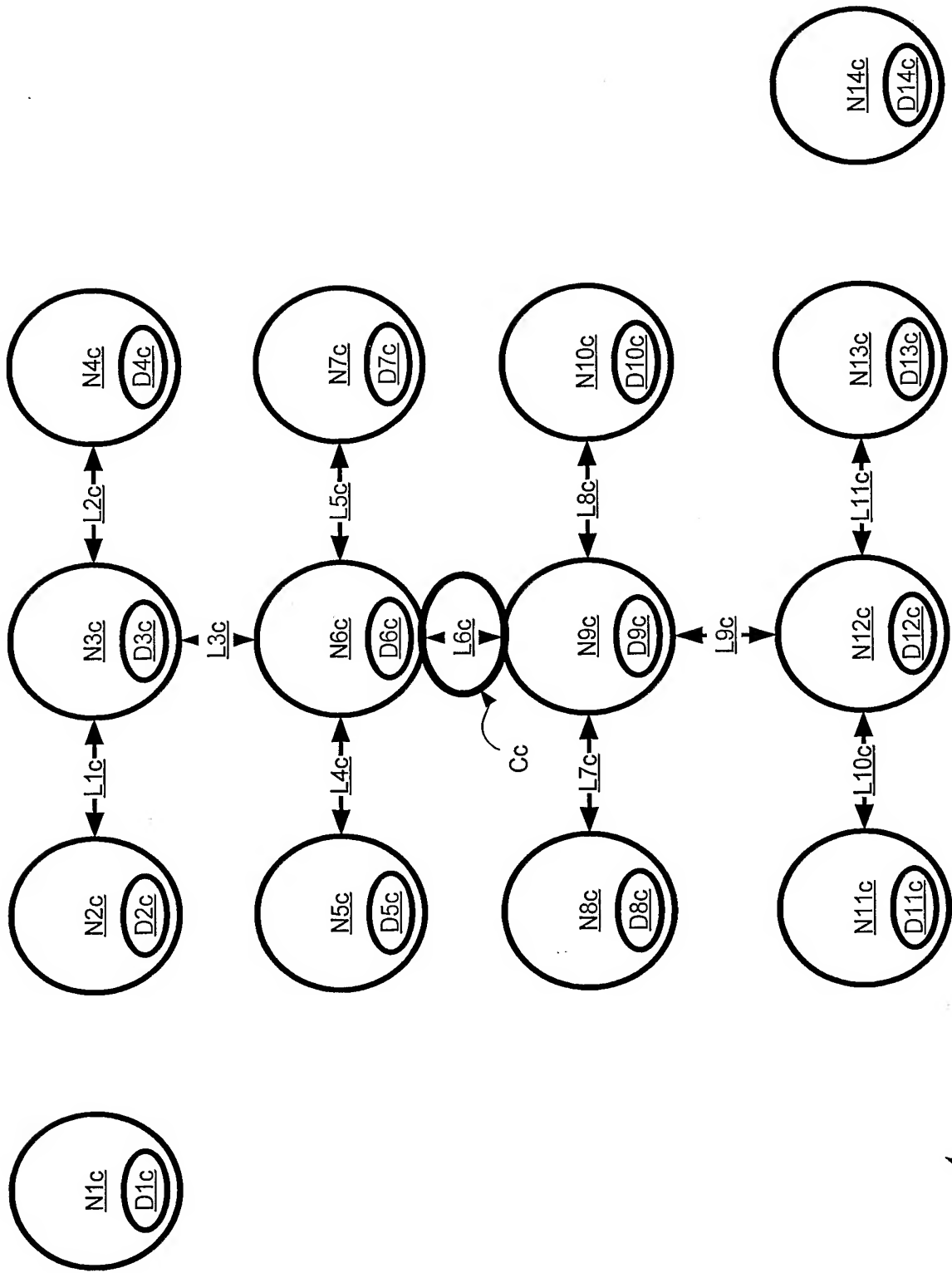


Fig. 14

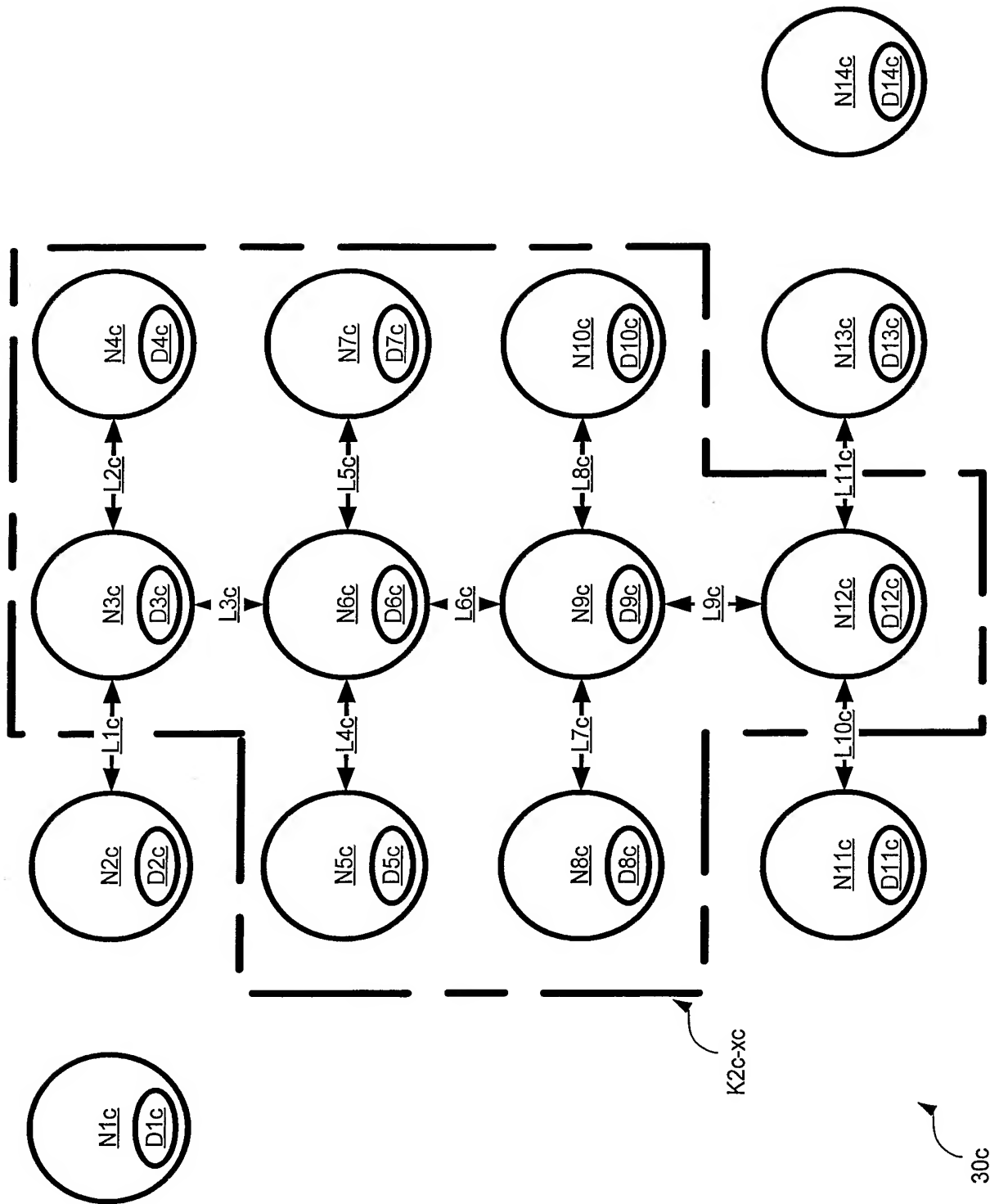


Fig. 15

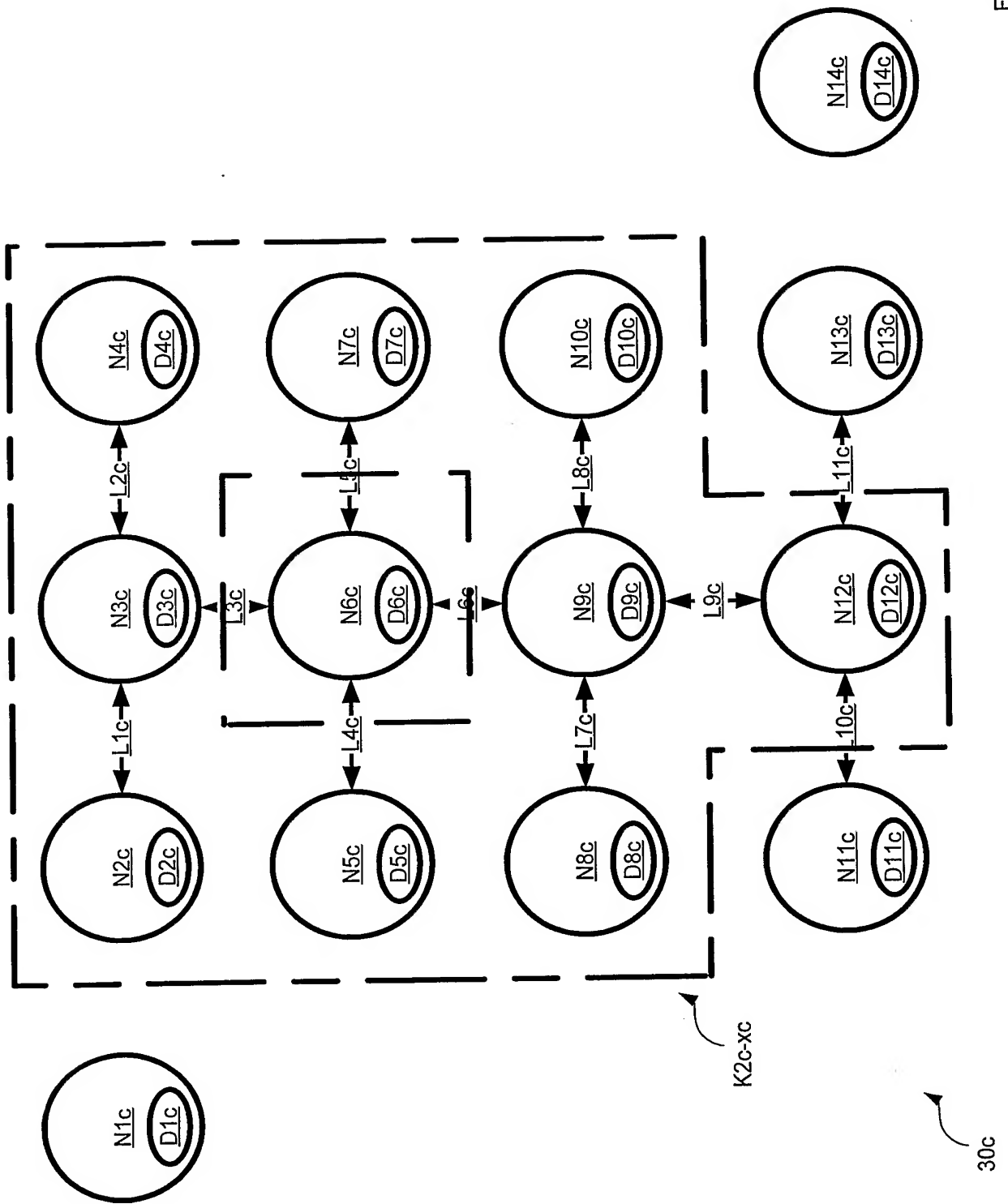


Fig. 16

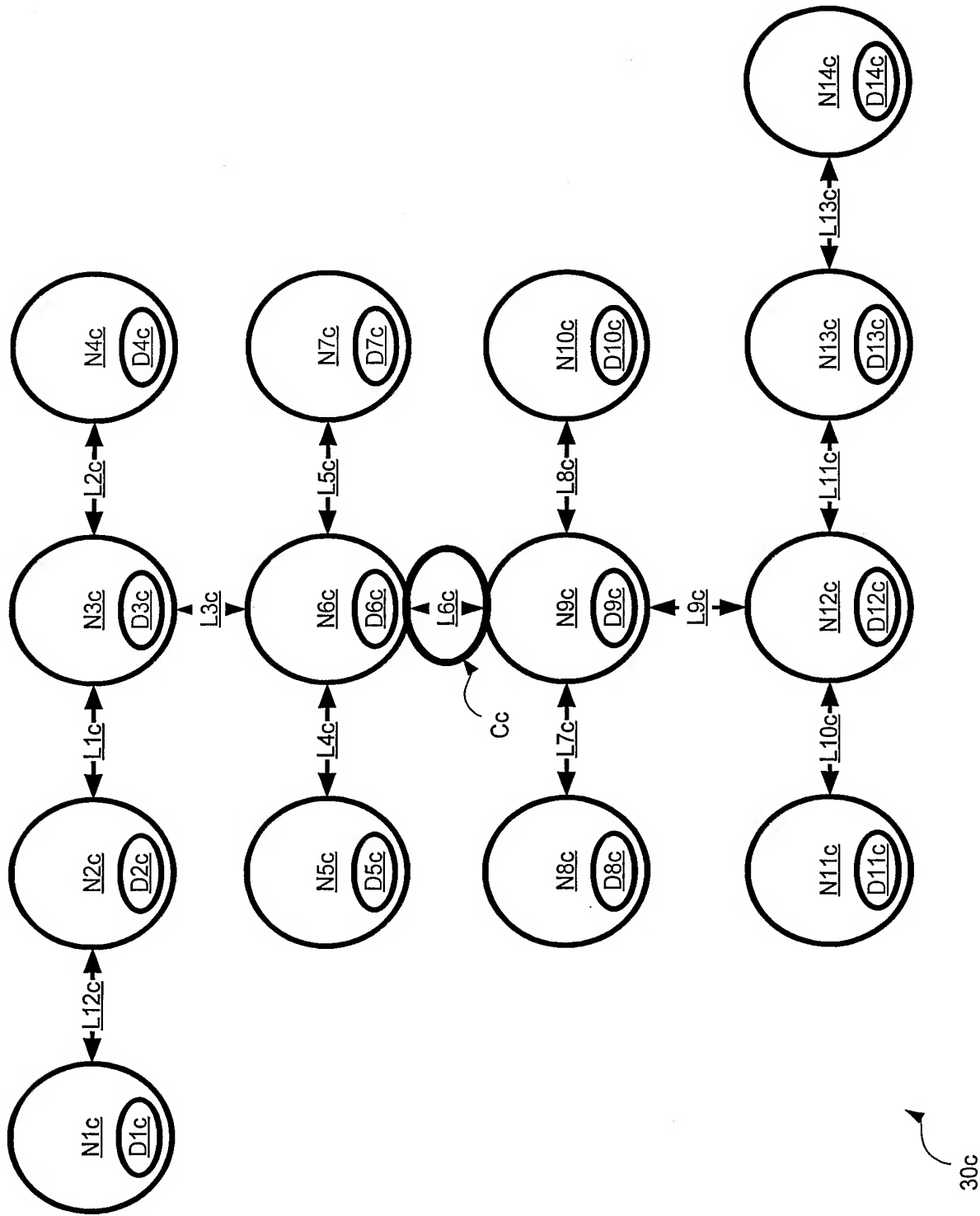


Fig. 14

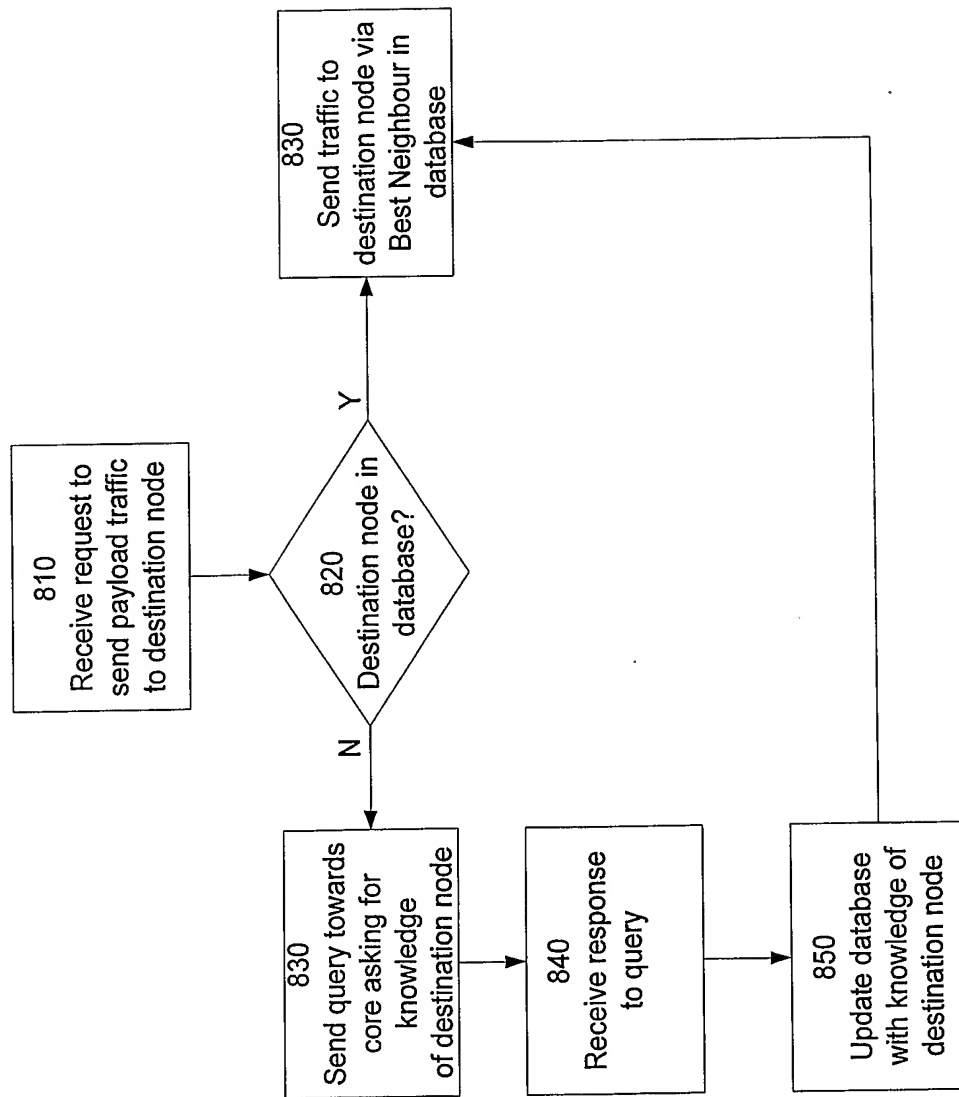


Fig. 18

800

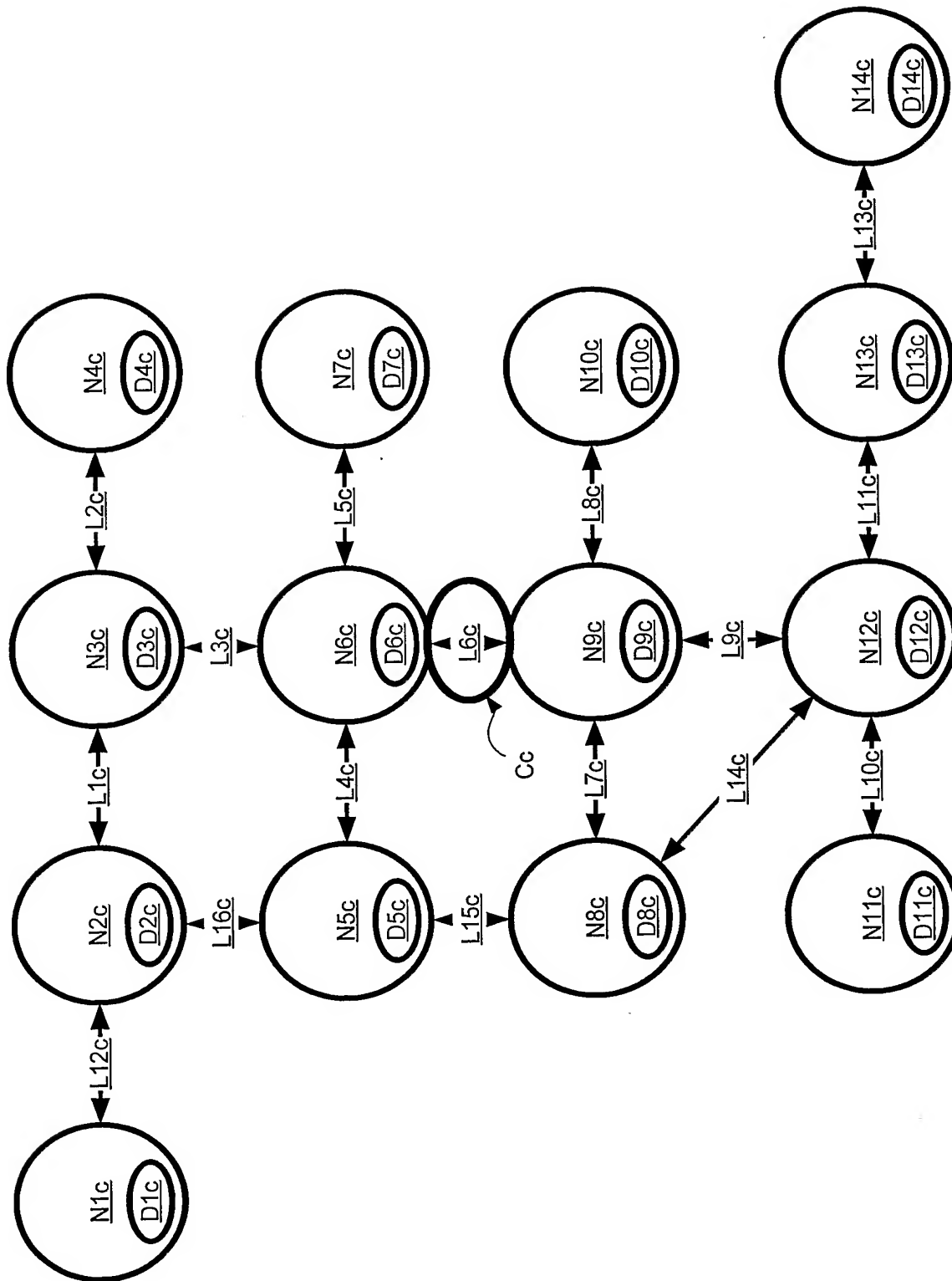


Fig. 19

30c

20/54

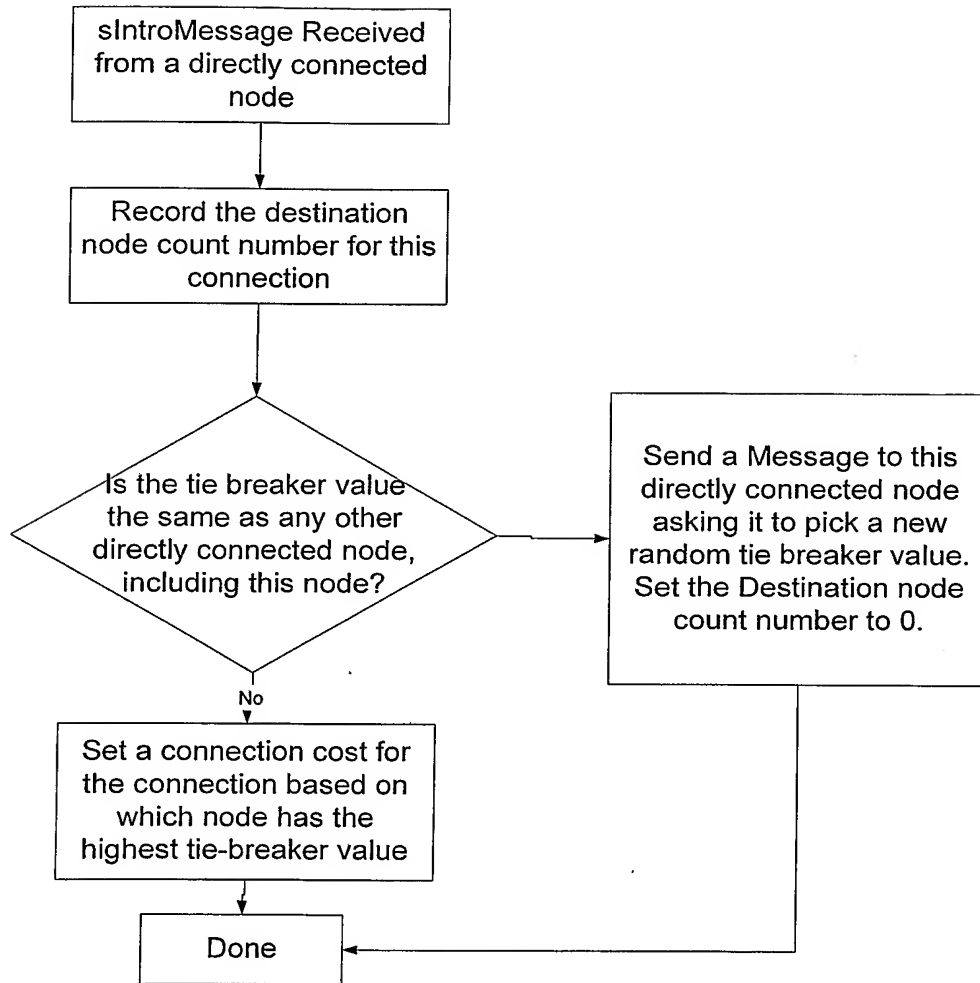


Fig 20

21/54

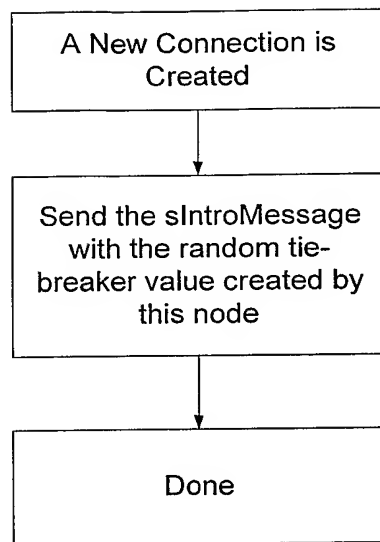


Fig 21

22/54

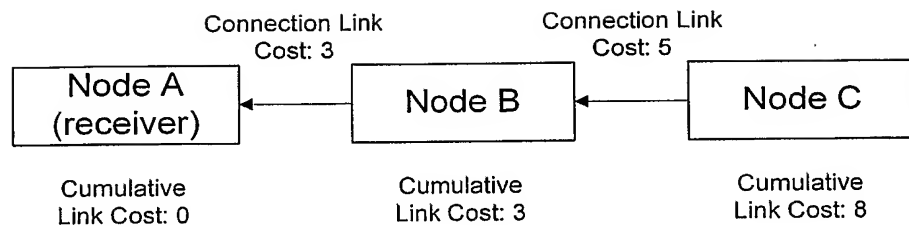
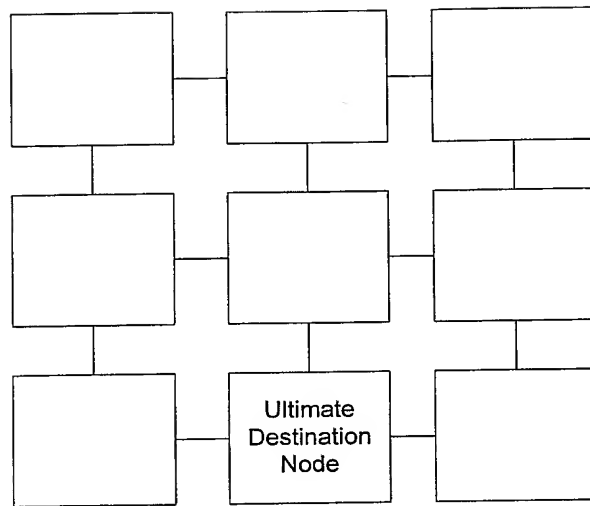


Fig 22

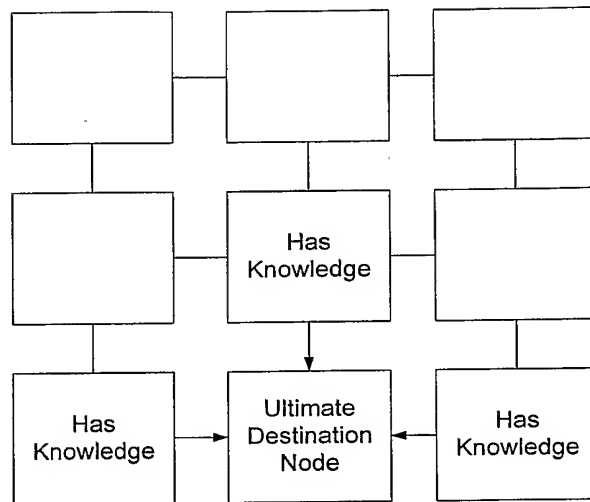
23/54



The ultimate destination node is connected to several nodes in the network and prepares to send an initial destination node update to its directly connected neighbor nodes.

Fig 23

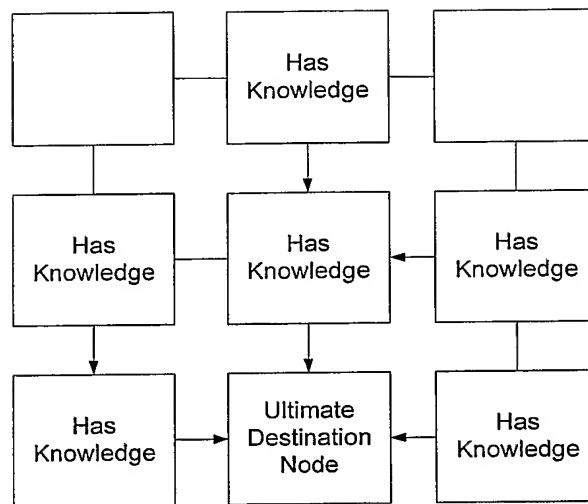
24/54



As destination node knowledge spreads, each node that gets told of the destination node selects the node that told it as the 'target node' for messages routed to that destination node.

Fig 24

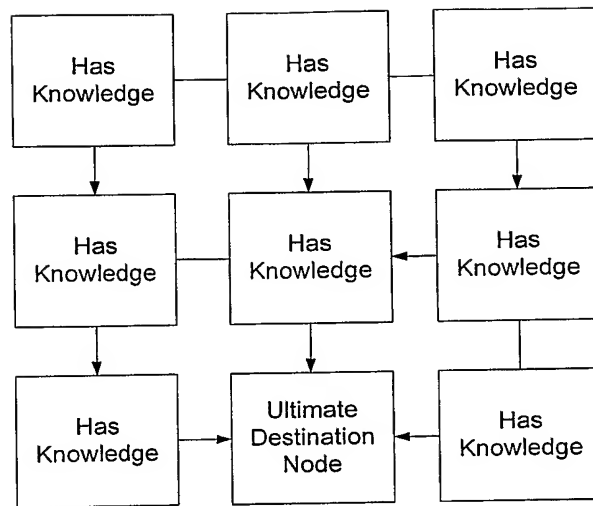
25/54



Knowledge of the destination node continues to spread, with each node that is told about the destination node choosing the node that told it. If another node provided it with a better hop cost later, the node would choose the directly connected node with the better hop cost.

Fig 25

26/54



In this step all nodes in the network are aware of the destination node and have a 'target node' that they can send messages that are destined for the destination node.

Fig 26

27/54

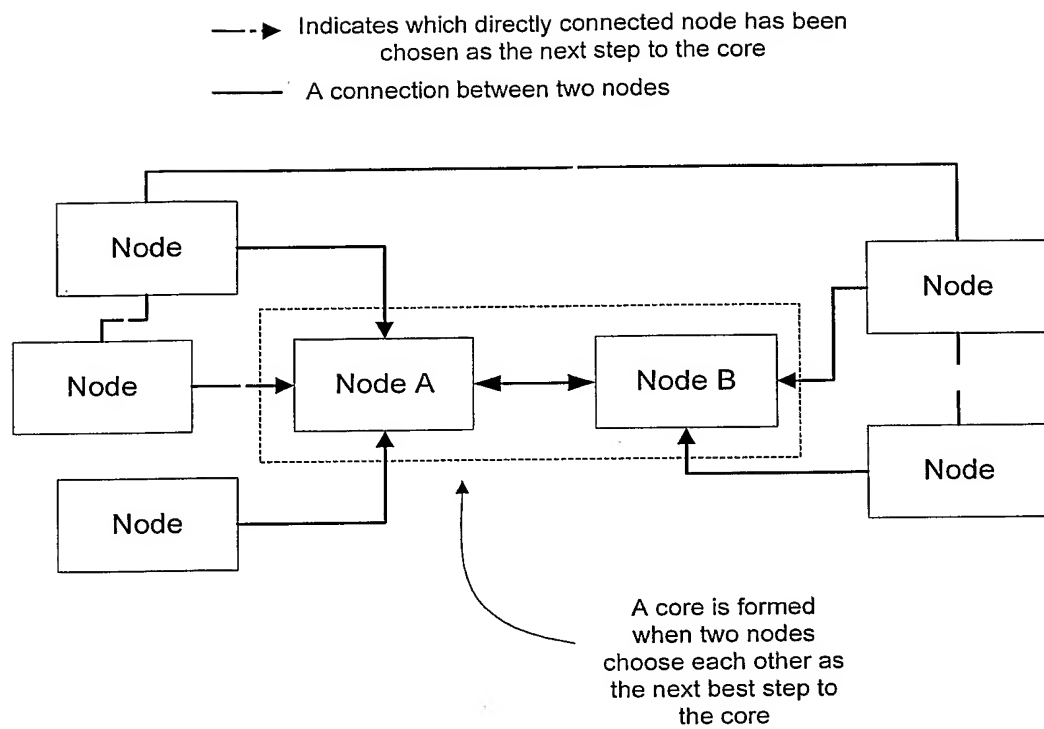


Fig 27

28/54

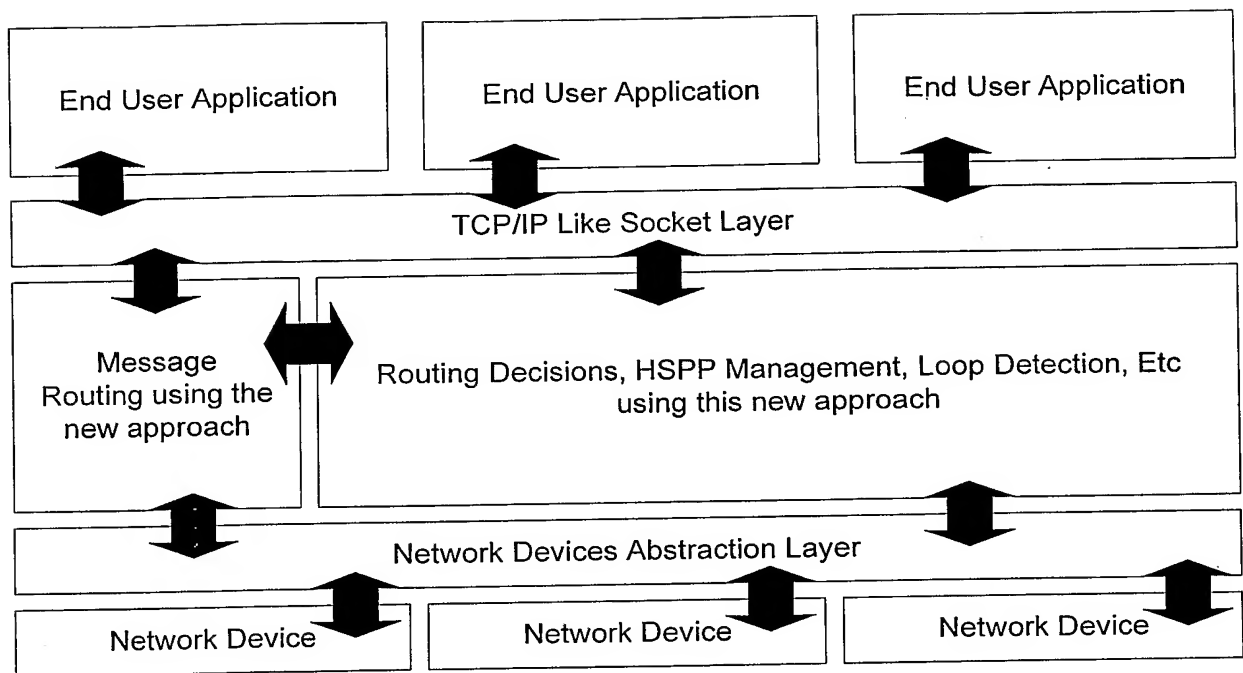


Fig 28

29/54

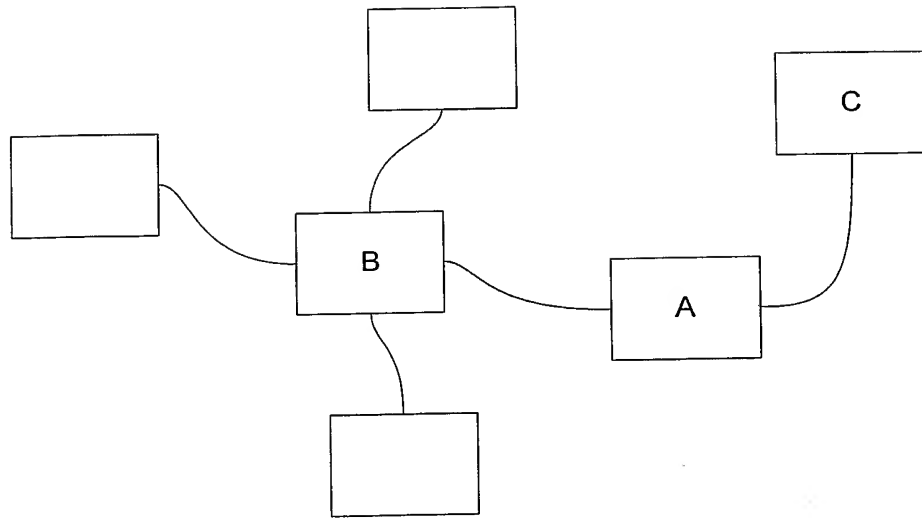


Fig 29

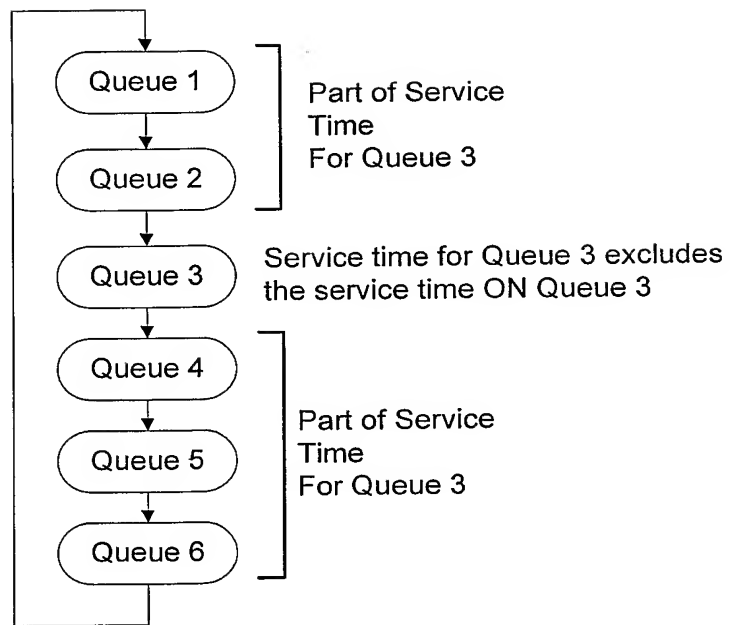


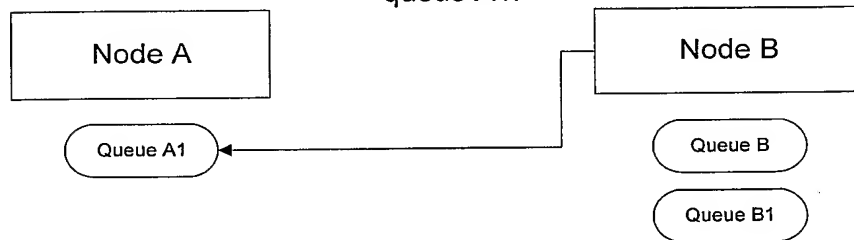
Fig 30

31/54

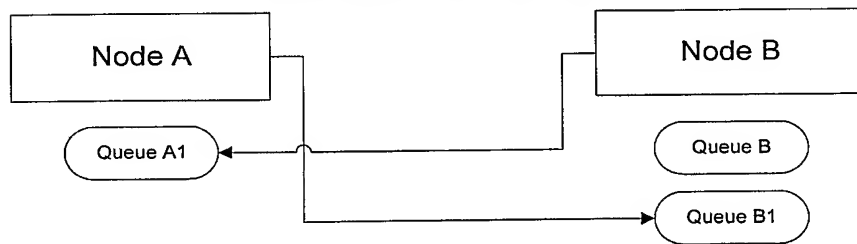
Step 1 – Node A creates queue A1 and asks node B For a queue to send messages to.



Step 2 – Node B creates queue B1 and tells node A about it using queue A1.



Step 3 – Node A sends a messages to queue B1 and node B sends ACK's to node A1.



Step 4 – Node A sends a messages to queue B confirming a connection to queue B1

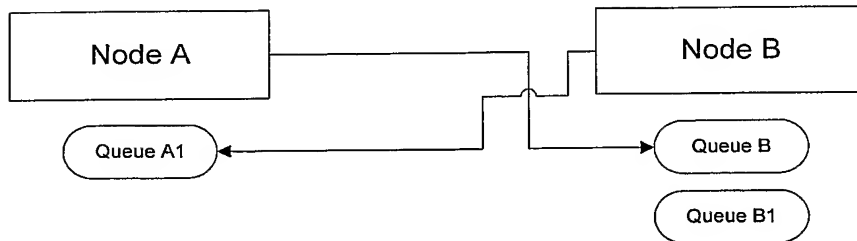
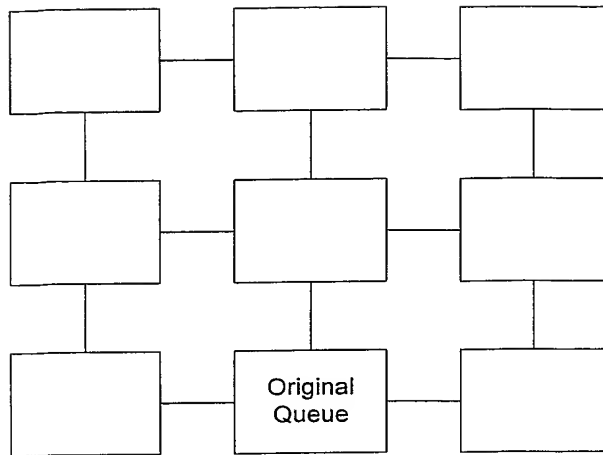
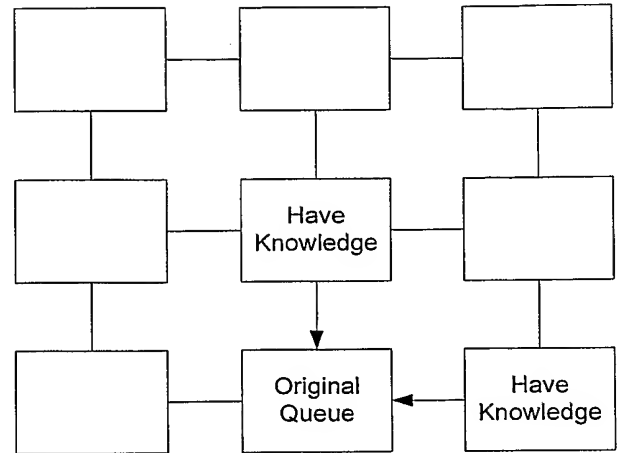


Fig 31

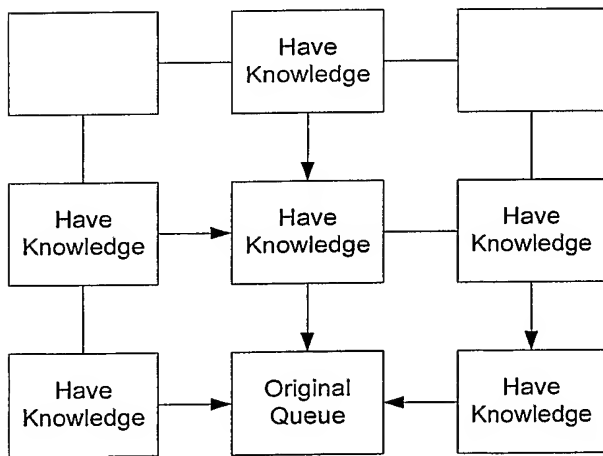
32/54



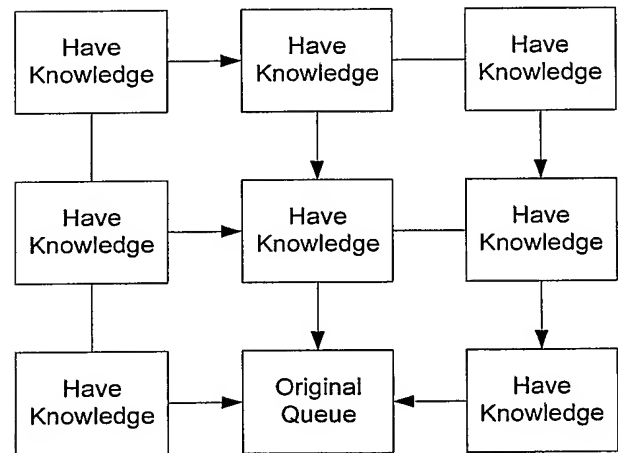
Step 1 – an EUS has created a Queue, and only the node that contains the EUS knows of the queue.



Step 2 – Next iteration, two directly connected nodes now know of the queue. The arrows represent chosen destinations.



Step 3 – Knowledge continues to spread



Step 4 – The whole network now has knowledge of the queue and there are no loops.

Fig 32

33/54

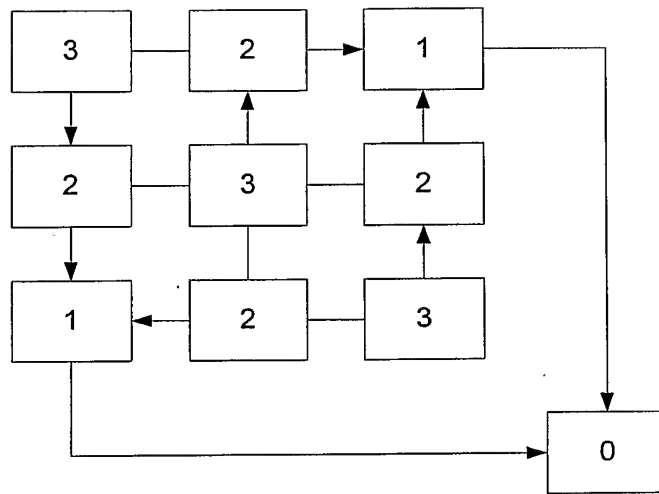


Fig 33

34/54

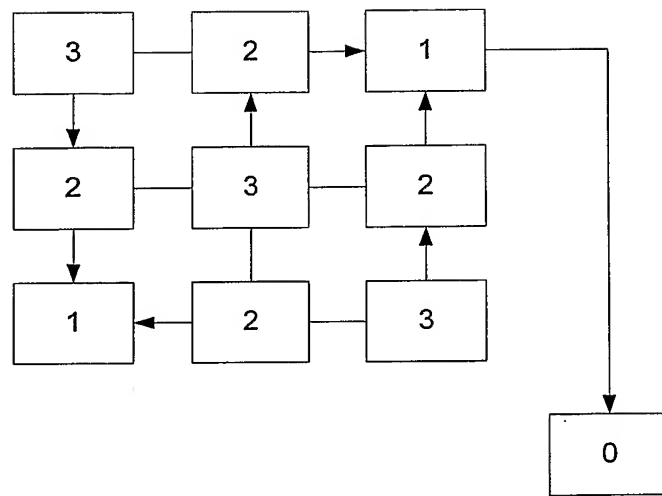


Fig 34

35/54

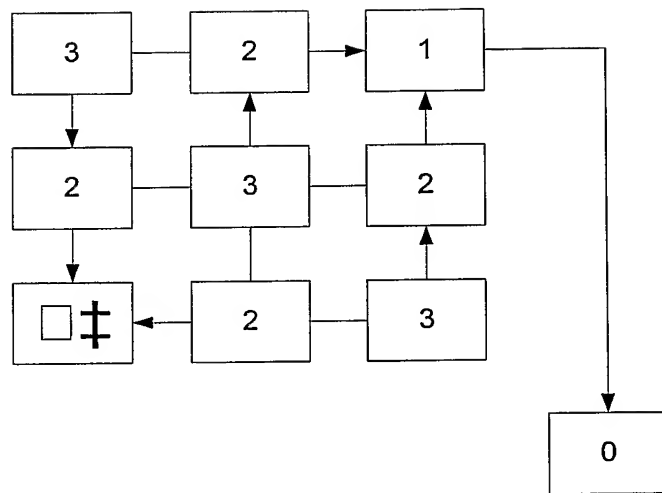


Fig 35

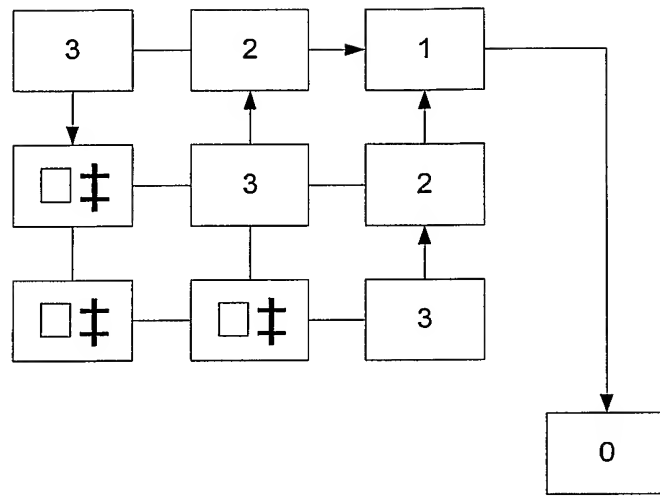


Fig 36

38/54

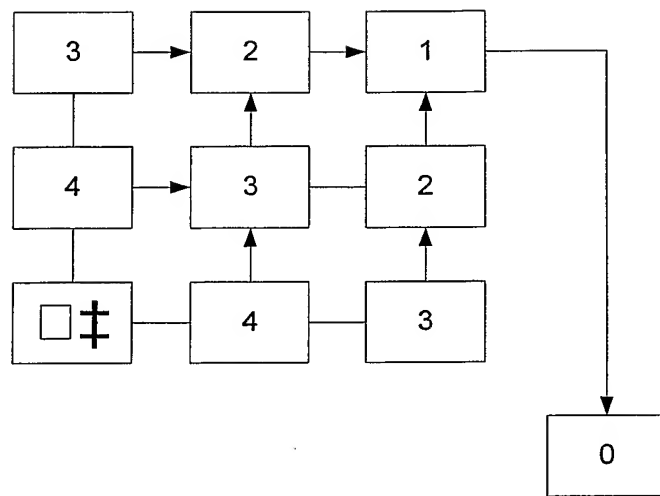


Fig 38

39/54

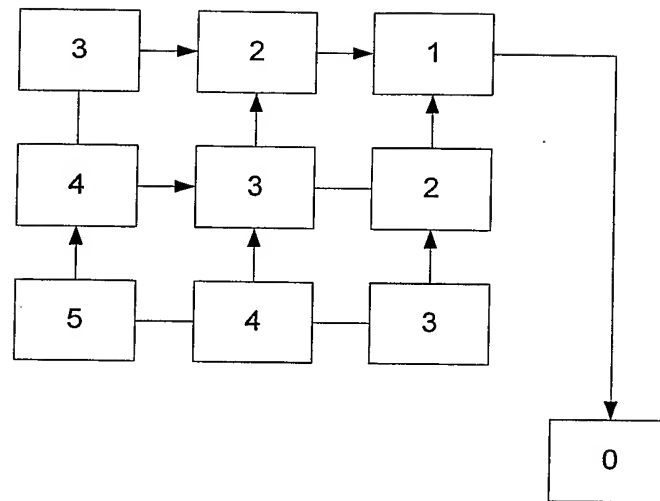


Fig 39

Incoming Latency Update

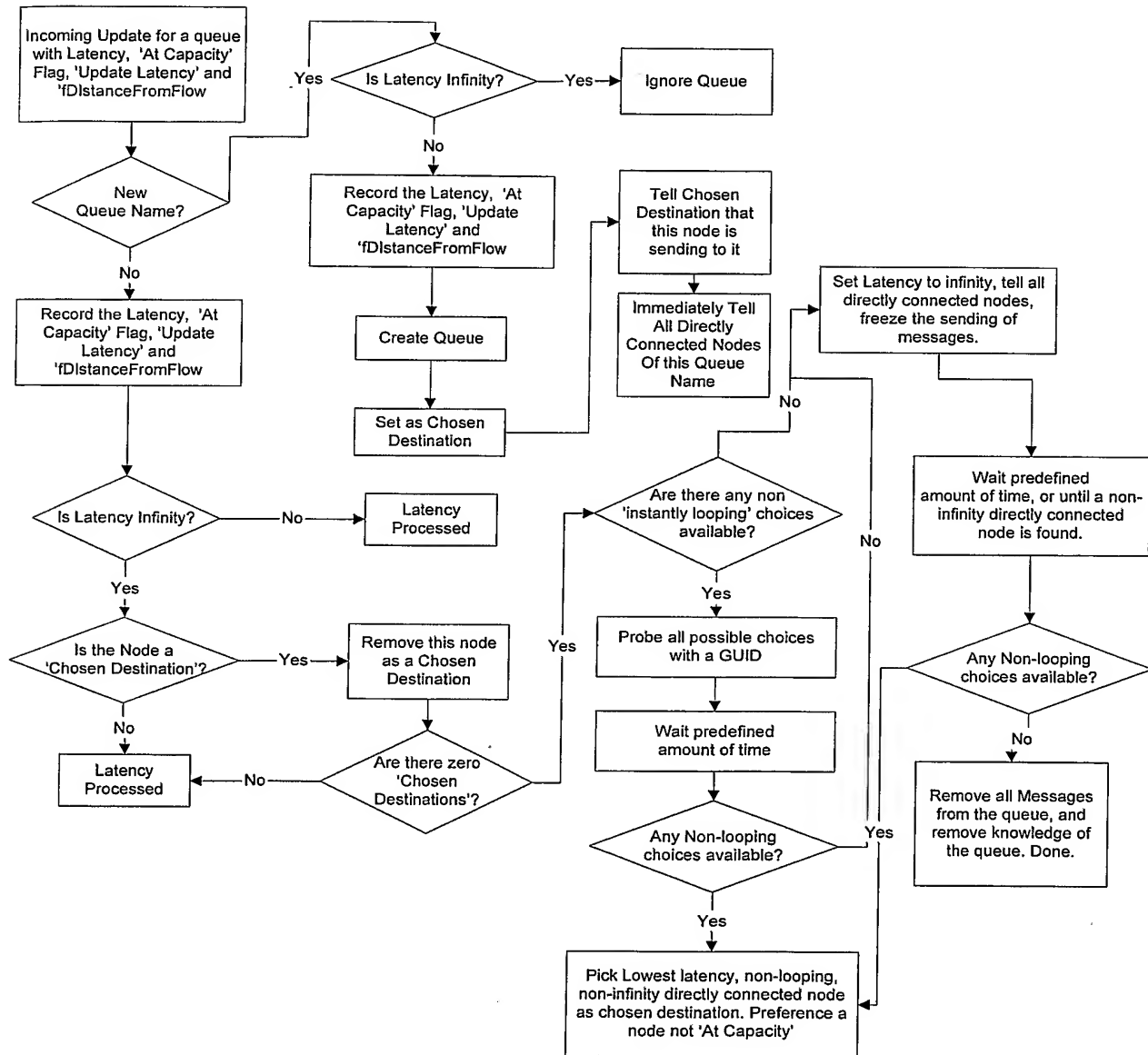


Fig 40

41/54

Latency At Infinity

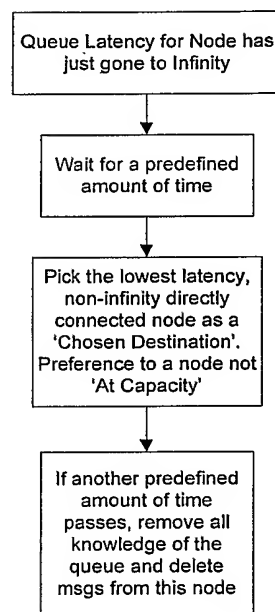
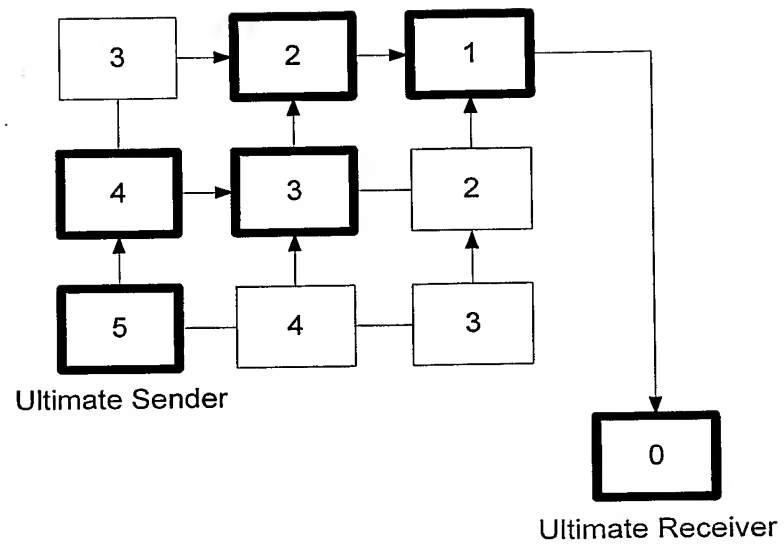


Fig 41

42/54



The nodes with the thick outlines constitute the data path

Fig 42

43/54

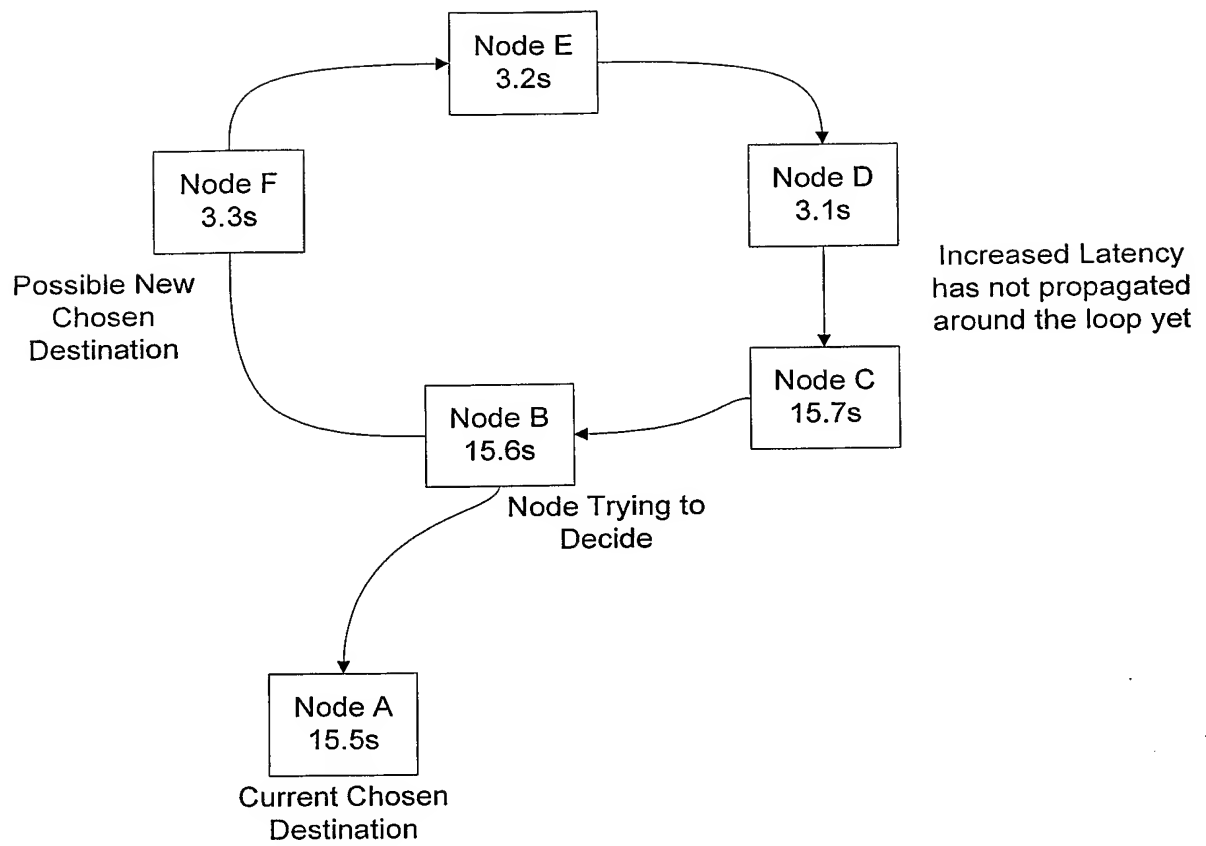


Fig 43

44/54

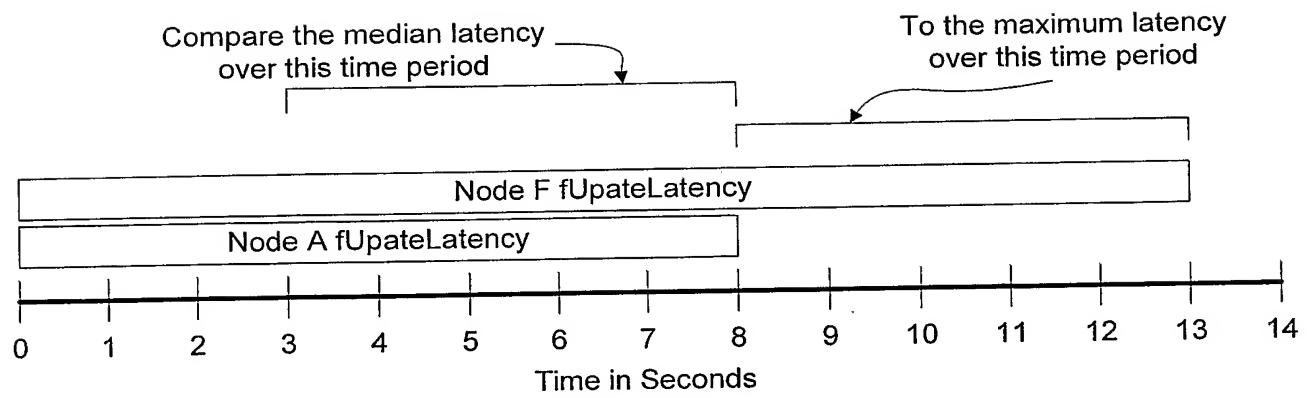


Fig 44

45/54

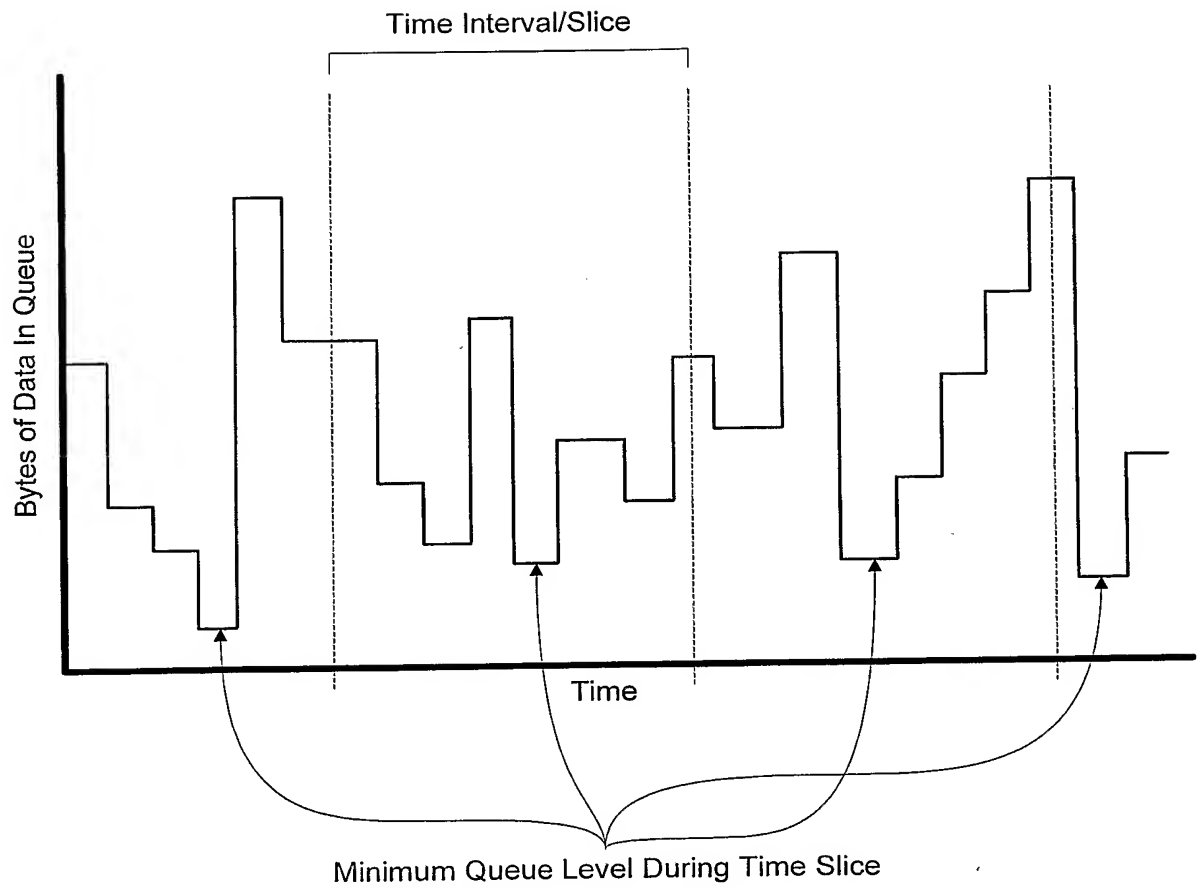


Fig 45

46/54

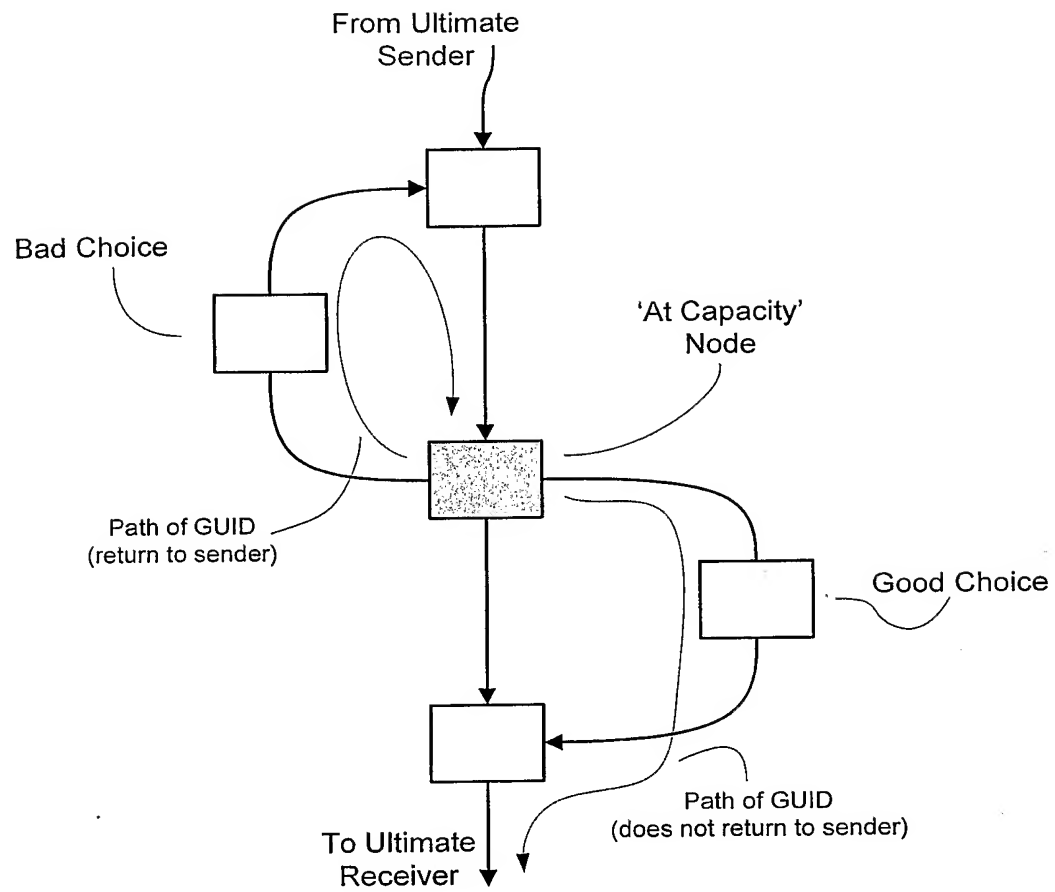


Fig 46

47/54

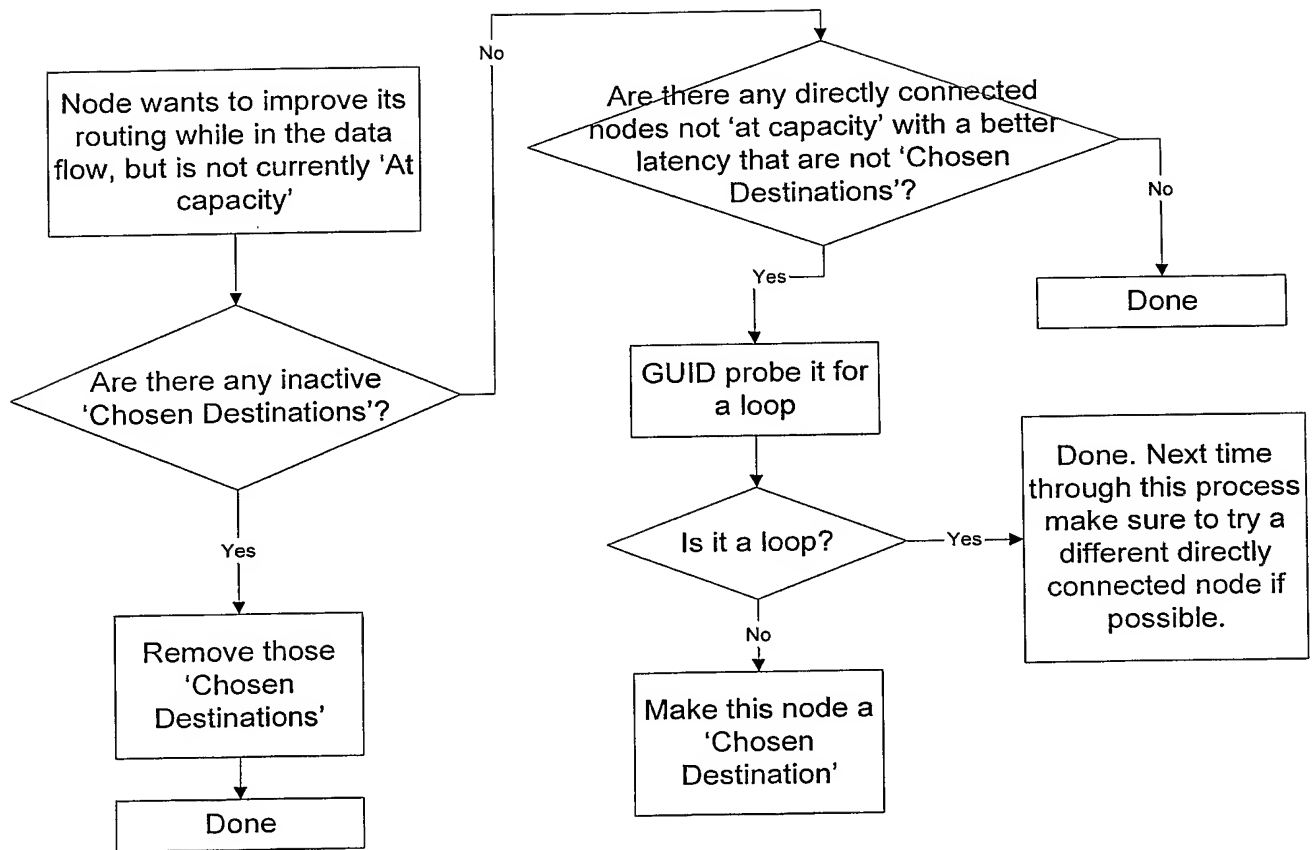


Fig 47

48/54

The latencies of these nodes are spiraling upwards because of the loop.
fUpdateLatency will also be spiraling up.

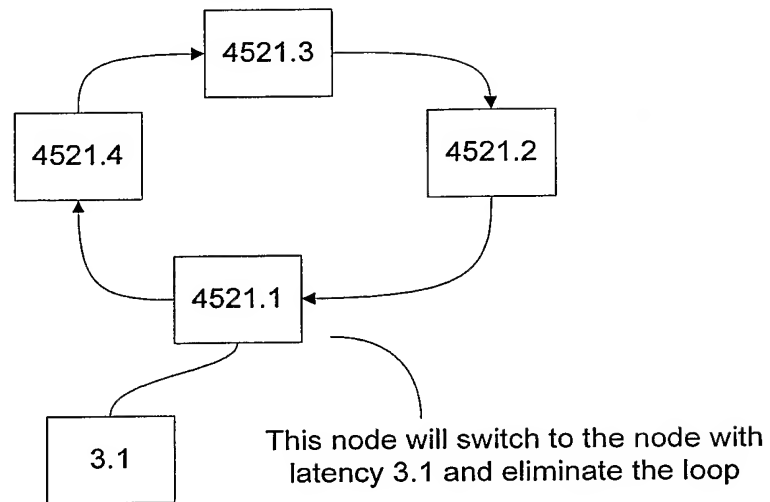


Fig 48

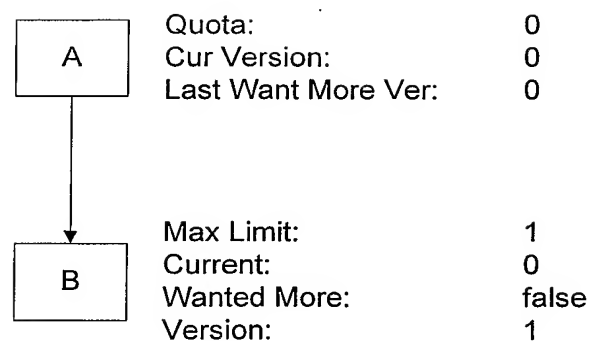


Fig 49

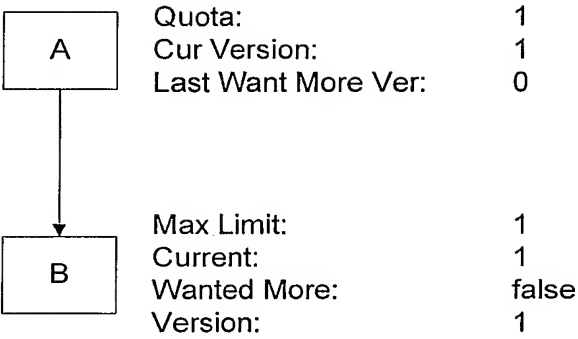


Fig 50

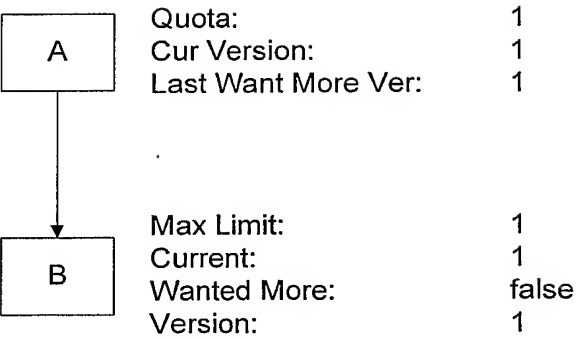


Fig 51

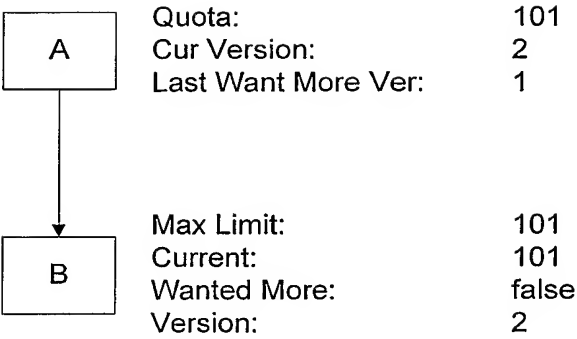


Fig 52

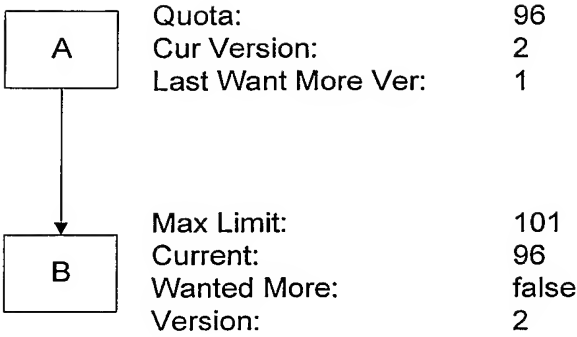
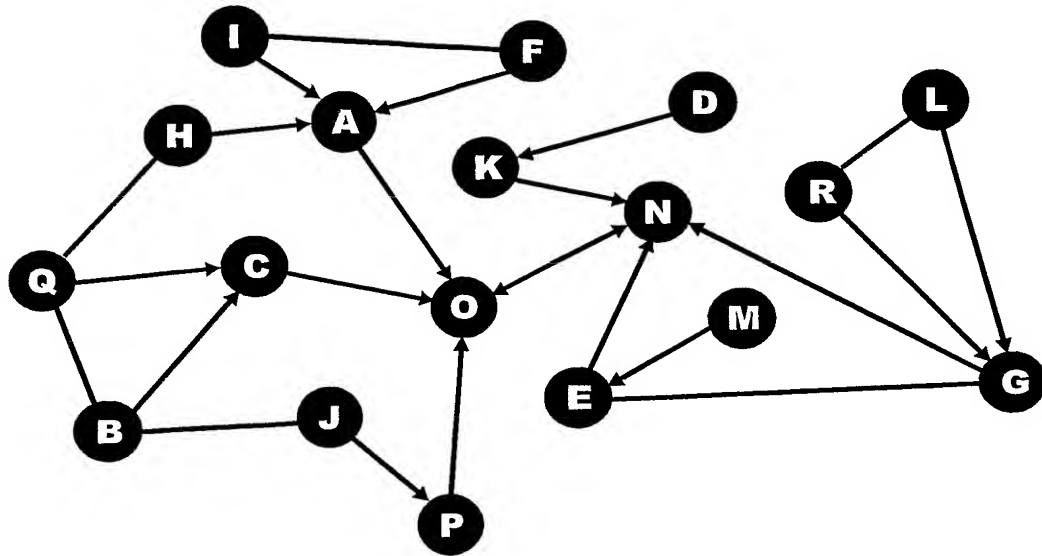


Fig 53

54/54

A network of nodes connected by links. An arrow indicates a nodes choice for its next best step to the core. A line with two arrows indicates two nodes have chosen each other as their next best step to the core.



This is the identical network that has been rearranged to better show the hierarchy that is formed by having each node pick its next best step to the core. The two nodes at the top of the hierarchy form the core.

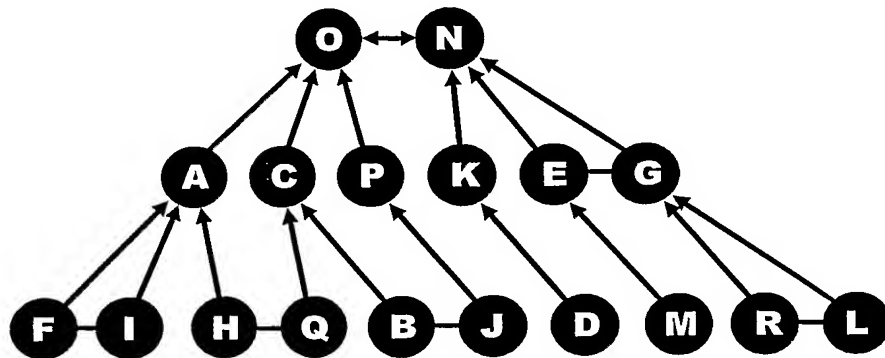


Fig 54